韓國經營科學會誌
第21巻 第1號
1996年 4月

대규모 비분리 콘벡스 최적화: 미분가능한 경우*

박구현* · 신용식**

Large-scale Nonseparabel Convex Optimization: Smooth Case*

Koohyun Park** · Young-Sik Shin**

Abstract

There have been considerable researches for solving large-scale separable convex optimization problems. In this paper we present a method for large-scale nonseparable smooth convex optimization problems with block-angular linear constraints. One of them is occurred in reconfiguration of the virtual path network which finds the routing path and assigns the bandwidth of the path for each traffic class in ATM(Asynchronous Transfer Mode) network. [1]

The solution is approximated by solving a sequence of the block-angualr structured separable quadratic programming problems. Bundle-based decomposition method[10,11,12] is applied to each large-scale separable quadratic programming problem. We implement the method and present some computational experiences.

1. 서 론

본 논문에서는 다음과 같은 선형의 블록-삼각 구조를 갖는 대규모 콘벡스 최적화 문제(LCOP: Large-scale Convex Optimization Problem)를 다룬다:

$$\text{Minimize } f(x_1, x_2, \cdots, x_N) \tag{1.1}$$

Subject to

$$\begin{array}{llll} B_1x_1 & & & = b_1 \\ & B_2x_2 & & = b_2 \\ & & \cdot & = \cdot \\ & & B_Nx_N & = b_N \end{array} \tag{1.2}$$

$$A_1x_1 + A_2x_2 + \cdots + A_Nx_N \le a$$

$$x_1 \ge 0,\ x_2 \ge 0,\ \cdots,\ x_N \ge 0$$

---

* 본 연구는 과학재단(941-1000-032-2) 지원에 의하여 이루어 졌음
** 홍익대학교 산업공학과

여기서 함수 $f:\mathrm{R}^{n_1+n_2+\cdots+n_N} \rightarrow \mathrm{R}$는 $x=(x_1, x_2, \cdots, x_N) \in R^{n_1+n_2+\cdots+n_N}$에 대해 콘벡스 (closed proper convex)이다. f는 분리가능(separable)일 필요가 없다. $a \in \mathrm{R}^m$이고, 각각의 $i=1, 2, \cdots, N$에 대해서 $b_i \in R^{m_i}$, $B_i \in R^{m_i \times n_i}$ 및 $A_i \in R^{m \times n_i}$ 이다. 통상적으로 결합조건식(*coupling constraints*)의 수는 m으로 분리조건식의 수인 $(m_1+m_2+ \cdots + m_N)$이나 문제의 차원인 $(n_1+n_2+ \cdots + n_N)$에 비해 현저히 적다.

대규모 최적화 문제는 생산일정 및 재고관리 문제, 생산일정 및 분배계획 문제, 산업간 입력-출력 분석, 다상품 네트워크 흐름(multicommodity network flow) 문제 등 많은 응용이 있으며, 목적함수 및 제약조건식이 분리가능하고 콘벡스인 경우의 해법에 대해서는 많은 연구가 있다. [3,7] 본 연구의 동기는 통신분야에서 BISDN의 ATM망 가상경로 재구성(Virtual Path Reconfiguration) 연구에서 비롯되었으며 가상경로 재구성문제가 (LCOP)의 구조를 갖는 비선형 목적함수의 다상품 네트워크 흐름 문제로 모형화 될 수 있다. [1] 가상경로 재구성 문제의 목적함수는 각 링크에서의 다양한 트래픽 클래스에 의한 트래픽 지연과 관련되어 트래픽 클래스별로 분리되지 않기 때문에 기존의 대규모 최적화 문제를 푸는 분해법(decomposition method)를 적용하는 데 어려움이 있다.

문제 (LCOP)의 목적함수 f가 선형인 경우, 즉 $f=\sum_{i=1}^{N} \langle c_i, x_i \rangle$이면 문제 (LCOP)는 Dantzig & Wolfe의 분해법 및 Rosen의 분할법(partitioning method) 등이 잘 알려져 있다. [3,7] 함수 f가 덧셈에 대해 분리 가능한 경우, 즉 $f(x_1, x_2, \cdots, x_N) = \sum_{i=1}^{N} \mathrm{f}_i(\mathrm{x}_i)$ 이면 (LCOP)는 Robinson[11,12]의 번들-분해법(BBD:bundle-based decomposition)이 매우 효과적이다. Medhi[10]는 목적함수 (1. 1)가 선형이고 제약조건식 (1. 2) 중 결합조건식이 등식일 때 번들-분해법을 MINOS 및 DECOMP(Dantzig-Wolfe 분해법의 구현)과 비교하여 다양한 크기 형태의 문제에 적용하여 번들-분해법의 우월성을 입증하였다. 번들-분해법이란 (LCOP)가 분리가능할 때 원 문제의 Rockafellar 쌍대(또는 conjugate 쌍대라고도 함, [13,14] 참고)를 구하면 목적함수가 비미분이 되는데 여기에 Lemarechal[8,9]의 번들법(bundle method)을 적용하는 것이다. 비미분인 쌍대문제의 함수값과 subgradient는 분할된 원 문제의 작은 문제의 해를 이용하여 계산된다.

(LCOP)와 같이 제약조건은 특별한 구조를 갖고 있으나 목적함수가 분리되지 않는 경우의 대규모 콘벡스 최적화 문제에 대한 효율적인 해법 연구는 잘 알려지지 않고 있다. 물론 제약식의 구조를 이용하지 않고 목적함수의 미분성에 따라 적절한 비선형 최적화 문제의 해법을 적용할 수 있겠다. 본 연구에서는 (LCOP)의 목적함수가 미분가능할 때 축차적으로 목적함수를 2차함수로 근사화시켜 번들-분해법을 적용하고자 한다. 이를 위해서 먼저 분리 가능한 2차 함수의 목적함수를 갖는 번들-분해법을 구현하였다. 이는 Medhi[10]가 분리가능한 선형 함수에 대해 번들-분해법을 구현한 연구와 비교될 수 있다.

본 논문의 구성은 제 2 절에서 비분리 대규모 콘벡스 최적화 문제인 (LCOP)의 해법으로 목적함수를 분리가능한 2차 함수로 근사화시켜 번들-분해법을 축차적으로 대입하는 SQA 알고리즘을 제시하며, 제 3 절에서는 SQA 알고리즘을 구현하고 제 4 절에서는 SQA 알고리즘의

수렴성을 검토하며 제 5 절에서는 SQA 알고리즘의 수치 적용을 소개한다. 마지막 절은 결론이다.

본 논문에서 사용되는 용어와 표현에 대해 간략히 소개한다. 먼저 R^n의 두 벡터 x와 y의 내적을 $\langle x, y\rangle$로 표현한다. x의 2-norm은 $\|x\|$로 표시하고 $\langle x, x\rangle^{\frac{1}{2}}$이다. 콘벡스 함수 f의 conjugate는 f^*로 표시하고

$$f^*(x) = \max\{\langle x, y\rangle - f(z) \mid z \in R^n\} \quad (1.3)$$

로 정의된다. 콘케이브 함수 g의 점 $y \in R^m$에서의 ε-subdifferential $\partial_\varepsilon g(y)$를

$$\partial\varepsilon g(y) = \{\pi \in R^m \mid \forall z, \; g(z) \le g(y) + \langle \pi, z-y\rangle + \varepsilon\} \quad (1.\ 4)$$

의 집합으로 정의하고 그 원소를 ε-subgradient라고 한다. (ε이 0일 때의 ε-subdifferential을 subdifferential이라고 하고 $\partial g(y)$로 표현한다.) 따라서 y^*가 함수 g의 ε-최대점이 되기 위한 필요 충분조건은 $0 \in \partial\varepsilon g(y^*)$이다. 그 밖에 용어 및 표현에 대해서는 [13]을 따른다.

2. SQA(Separable Quadratic Approximation) 알고리즘

번들-분해법[11,12]을 축차적으로 이용하여 (LCOP)의 해를 구하는 SQA 알고리즘을 제시한다. (LCOP)에 Rockafellar의 쌍대[14]를 적용하면 쌍대문제는 다음과 같이 된다.

$$\text{Maximize } \{g(y) \mid y \ge 0\} \quad (2.1)$$

여기서

$$\begin{aligned} g(y) &= \text{Min } f(x_1, x_2, \cdots, x_N) + \langle \sum_{i=1}^{N} A_i x_i - a, y\rangle \\ &\quad \text{s/t } B_i x_i = b_i,\ x_i \ge 0,\ i=1,2,\cdots,N \qquad (2.2) \\ &= -\langle a, y\rangle + \left\{ \begin{array}{l} \text{Min } f(x_1, x_2, \cdots, x_N) + \langle \sum_{i=1}^{N} A_i x_i, y\rangle \\ \text{s/t } B_i x_i = b_i,\ x_i \ge 0, \\ \qquad i=1,2,\cdots,N \end{array} \right. \end{aligned}$$

함수 $g(y)$에서 $f(x_1, x_2, \cdots, x_N)$가 분리가능하지 않기 때문에 번들-분해법에서와 같이 분할된 부문제로 해결되지 않는다.

SQA법은 함수 $f(x_1, x_2, \cdots, x_N)$를 분리가능한 2차함수로 근사화시켜 번들-분해법을 적용하여 보다 개선된 해를 얻어내는 방법이다. 즉 현재해 $x^k=(x_1^k, x_2^k, \cdots, x_N^k)$에서 함수 f를 2차적으로 근사화시키고, Hessian 행렬을 블록-대각 행렬로 근사화시킨다. 즉

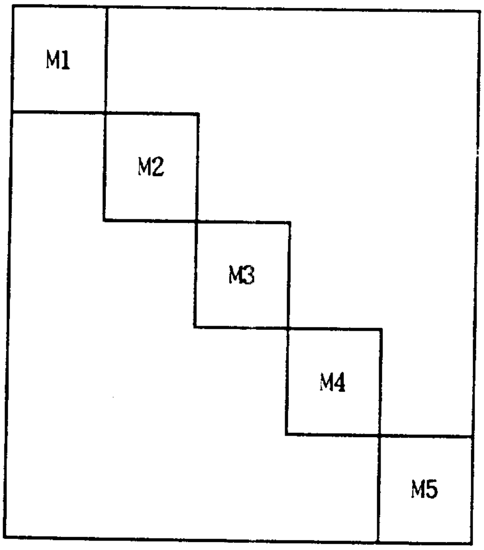
$$\begin{aligned} f(x) &\approx \frac{1}{2}\langle \nabla^2 f(x^k)(x-x^k), x-x^k\rangle + \langle \nabla f(x^k), x-x^k\rangle + f(x^k) \\ &\approx \frac{1}{2}\sum_{i=1}^{N}\langle M_i(x_i-x_i^k), x_i-x_i^k\rangle + \sum_{i=1}^{N}\langle q_i(x_i-x_i^k\rangle + f(x^k) \qquad (2.3) \\ &= \frac{1}{2}\sum_{i=1}^{N}\langle M_i x_i, x_i\rangle + \sum_{i=1}^{N}\langle q_i - M_i x_i^k, x_i\rangle + \sum_{i=1}^{N}\xi_i + f(x^k) \end{aligned}$$

여기서

$$M_i = \frac{\partial^2 f(x^k)}{\partial x_i},\quad q_i = \frac{\partial f(x^k)}{\partial x_i},$$

$$\xi_i = \frac{1}{2}\sum_{i=1}^{N}\langle M_i x_i^k, x_i^k\rangle - \sum_{i=1}^{N}\langle q_i\ x_i^k\rangle \quad (2.4)$$

이다. [그림 1]은 함수 $f(x)$의 x^k에서의 gradient 벡터와 근사화된 Hessian 행렬를 나타낸다.



x^k에서의 $f(x)$의 gradient 벡터

q1	q2	q3	q4	q5

[그림 1] 함수 $f(x)$의 x^k에서의 gradient 벡터와 근사화된 Hessian 행렬

M_i와 q_i, ξ_i가 주어질 때, x_i^k에서 $f_i(x_i)$를 다음과 같이 정의한다.

$$f_i(x_i) = \begin{cases} \frac{1}{2}\langle M_i x_i,\ x_i\rangle \\ +\ \langle q_i - M_i x_i^k,\ x_i\rangle\ +\ \xi_i \\ \quad \text{만약 } B_i\ x_i = b_i \text{ 및 } x_i \geq 0 \text{이면} \\ \infty, \text{ 그렇지 않으면} \end{cases} \tag{2.5}$$

[정리 3] $f(x_1,\ x_2,\ \cdots,\ x_N)$가 콘벡스이면 함수 $f_i(x_i)$도 콘벡스이다.

(증명) 함수 f의 Hessian 행렬은 positive semi-definite이므로

$$(0,\ x_i^T,\ 0)\ \nabla^2 f(x)\ (0,\ x_i^T,\ 0)^T$$

$$= \langle M_i x_i,\ x_i\rangle\ \geq\ 0 \tag{2.6}$$

이다. 따라서 M_i는 positive semi-definite이고 $f_i(x_i)$는 콘벡스이다. ■

$x^k = (x_1^k,\ x_2^k,\ \cdots,\ x_N^k)$에서 2차식으로 근사화시킨 f를 쌍대문제의 목적함수 $g(y)$에 대입하면

$$g(y)\ \approx\ -\langle a,\ y\rangle + \sum_{i=1}^{N} \{\text{Minimize}$$

$$\{f_i(x_i) - \langle -A_i^T\ y,\ x_i > \}$$

$$= -\langle a,\ y\rangle - \sum_{i=1}^{N} f_i^*(-A_i^T\ y) \tag{2.7}$$

Subdiffeential $\partial g(y)$는

$$\partial g(y) =\ -\mathrm{a} + \sum_{i=1}^{N} \mathrm{A}_i\ \partial \mathrm{f}_i^*(-\mathrm{A}_i^T \mathrm{y}) \tag{2.8}$$

이 되고, $\partial f^*_i(-A_i^T\ y)$ = argmin $\{f_i(x_i) - \langle -A_i^T\ y,\ x_i\rangle\}$이다. 즉 $\partial f^*_i(-A_i^T\ y)$ 는 분할된 i번째 부문제인 2차계획문제 ($QP(y)$)의 해집합이 된다.

$$QP(y)_i \left\{ \begin{array}{l} \text{Minimize } \frac{1}{2}\langle M_i x_i,\ x_i\rangle \\ \qquad +\langle q_i - M_i x_i^k + A_i^T y,\ x_i\rangle \\ \text{Subject to } B_i x_i = b_i,\ x_i \geq 0 \end{array} \right.$$

(LCOP)의 해를 구하는 SQA 알고리즘을 요약하면 [그림 2]와 같다.

3. SQA 알고리즘의 구현

제 2 절에서 제시한 요약 SQA 알고리즘은 크게 두 부분으로 나누어 구현되었다. 첫부분은 Medhi[10]가 블록-삼각의 선형등식의 구조화된 제약조건을 갖는 선형계획문제를 위해 구현한 번들-분해법을 분리가능한 2차계획 문제에 적용될 수 있도록 수정한 것이다. 본 연구에서 구현한 2차계획문제의 제약식은 블록-삼각의 선형조건식이 등식일 때만이 아니라 부등식일 경우에 대해서도 함께 구현하였다. 제 2 절에서는 결합조건식이 부등식인 경우에 대해 번들-분해법을 전개하였다. 따라서 기존의 번들-분해법과 관련된 연구[10,11,12]가 등식의 결합조건식인 경우에만 해당되기 때문에 본 연구는 분리가능한 문제 측면에서 볼 때에는 번들-분해법을 부등식의 결합조건식을 갖는 2차계획문제로 확장한 구현 연구가 된다. 결합조건식이 부등식인 경우는 전절에서 보듯이 쌍대문제에 비음조건이 첨가된 비미분 최적화 문제가 된다. 따라서 방향벡터는 subgradient을 R^m_+에 투사시켜 계산해야 하는 등 번들이용 및 유지-변경이 달라진다. 파라미터, 변수 및 번들들의 표현은 Medhi의 표현 그대로 사용한다. 다만 분할된 원문제의 부문제는 선형계획문제 대신 2차계획문제를 선형 보완문제(linear complementary problem)으로 전환하여 Lemke 알고리즘을 적용하여 해를 구한다.

둘째 부분은 비분리인 (LCOP)의 목적함수를 분리가능한 2차 함수로 근사화시켜 번들-분해법을 적용할 수 있도록 하는 부분과 수렴성과 관련된 선형 탐색(line search) 부분이다. (LCOP)의 목적함수가 콘벡스이기 때문에 분할된 부문제는 콘벡스의 2차계획 문제이므로 Lemke 알고리즘을 적용할 때 유한번의 반복수로 해를 구할 수 있다. 수렴성과 관련된 선형 탐색은 국소수렴(local convergence)인 경우와 전체수렴(global convergence)인 경우로 나누어 진다.

다음의 각 step은 구현을 위한 단계이다.

(step 0) 초기화

1) 원문제 시작점 $\hat{x} = (\hat{x}_1, \hat{x}_2, \cdots, \hat{x}_N)$을 선택한다.

2) 쌍대문제 시작점 y^1를 선택한다.

3) 원문제 허용오차 τ를 선택한다.

4) 쌍대문제 허용오차 ε, $\delta > 0$, $\hat{\varepsilon} \geq \varepsilon$를 선택한다.

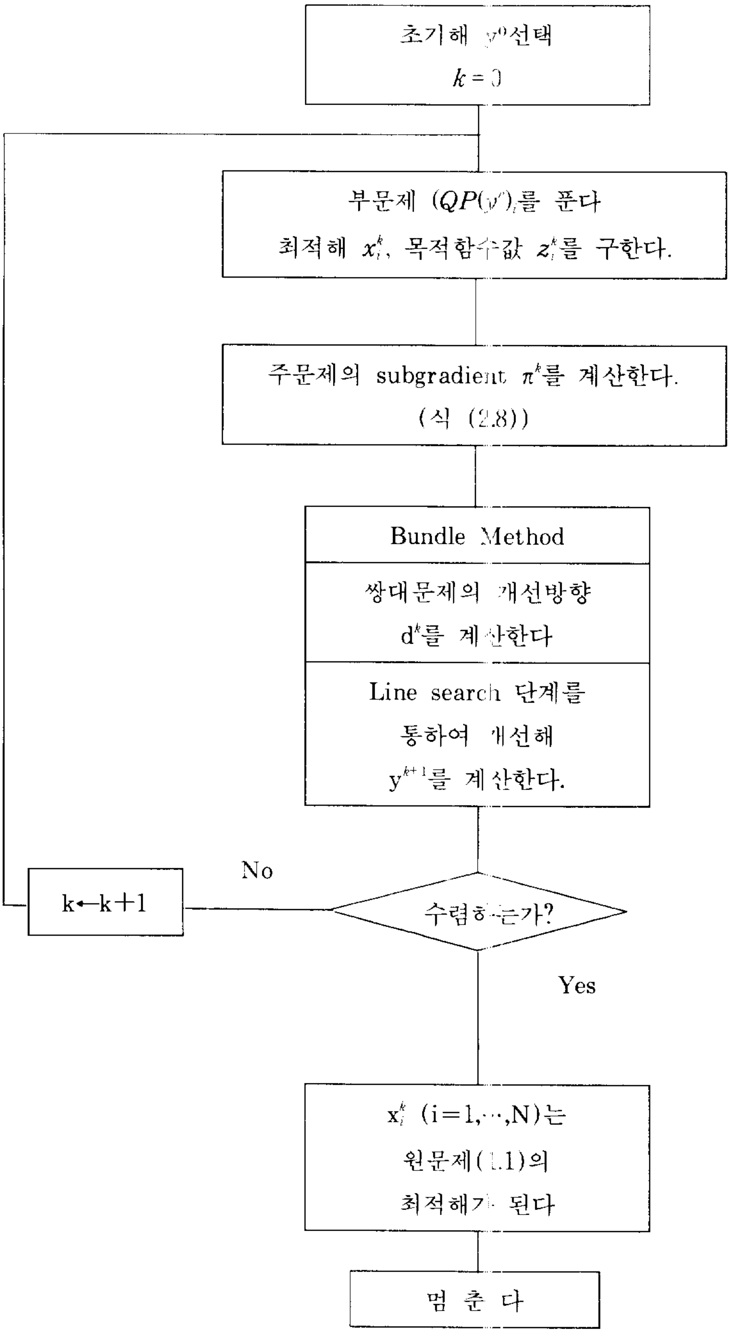
5) 최대 번들 크기 b를 선택한다.

6) 파라미터 β, r, μ $(0 < r < \beta < 1,\ \beta + \mu < 1,\ \mu > 0)$ 를 선택한다.

(step 1) SQA 근사화

1) $\hat{x}$에서 각 $i = 1,2,\cdots,N$에 대해

$$M_i = \frac{\partial^2 f(\hat{x})}{\partial x_i},\ q_i = \frac{\partial\ f(\hat{x})}{\partial\ x_i^2},$$



[그림 2] SQA 알고리즘 순서도

$\xi = \frac{1}{2} \sum_{i=1}^{N}$

$\langle M_i \hat{x}_i, \hat{x}_i \rangle - \sum_{i=1}^{N} \langle q_i, \hat{x}_i \rangle$를 계산한다.

(step 2) 분할된 부문제

1) y_1에서 N개로 분할된 부문제 $(QP(y)_i)$를 Lemke 알고리즘을 이용하여 푼다.

$$(QP(y)_i) \begin{cases} \text{Minimize } \frac{1}{2} \langle M_i x_i, x_i \rangle \\ \qquad + \langle q_i - M_i \hat{x}_i + A_i^T y, x_i \rangle \\ \text{Subject to } B_i x_i = b_i, \ x_i \geq 0 \end{cases}$$

2) $(QP(y^1)_i)$의 최적해를 x_i^1로, 최적 목적함수 값을 z_i^1로 놓는다.

(step 3) 쌍대문제 함수값 및 subgradient 계산

1) $g(y^1) = -\langle a, y^1 \rangle + \sum_{i=1}^{N} z_i^1$

2) $\pi^1 = -a + \sum_{i=1}^{N} A_i x_i^1$

(step 4) 초기번들 설정

1) $D \leftarrow \{\pi^1\}$, $P \leftarrow \{(x_1^1, x_2^1, \cdots, x_N^1)\}$, $K \leftarrow \{1\}$, $a_{11} = 0$

2) $k \leftarrow 1$, $\varepsilon^1 \leftarrow \hat{\varepsilon}$

⟨step 5⟩ 쌍대문제의 방향벡터 계산

1) $\varepsilon_k \leftarrow \max \{\varepsilon, \min\{\varepsilon_k, \hat{\varepsilon}\}\}$

2) 쌍대문제 해의 개선방향을 결정하는 2차 계획 문제 (DQP)를 Lemke 알고리즘을 이용하여 해를 구한다.

$$(DQP) \begin{cases} \min \frac{1}{2} \| \sum_{j \in K} \lambda_j (\pi^j)_+ \|^2 \\ s/t \ \sum_{j \in K} \lambda_j = 1 \\ \sum_{j \in K} \alpha_{jk} \lambda_j \leq \varepsilon_k \\ \lambda_j \geq 0, \ j \in K \end{cases}$$

여기서 $(\pi^j)_+$ 는 π^j를 R_+^m에 투사시킨 것이다.

3) (DQP)를 최적해를 $\lambda_j^k (j \in K)$로 놓고, s_k를 $\sum_{j \in K} \alpha_{jk} \lambda_j \leq \varepsilon_k$의 승수(mutiplier)로 놓는다. (DQP)를 선형보완문제로 전환시켜 Lemke 알고리즘을 적용하면 해와 함께 각 식의 승수도 함께 구해진다.

4) $d^k = \sum_{j \in K} \lambda_j^k (\pi^j)_+$를 계산한다.

5) $v_k = \|d^k\|^2 + s^k \varepsilon^k$를 계산한다.

(step 6) 번들-분해법의 수렴성 검사

1) $\|d^k\| < \delta$이며 $\sum_{j \in K} \alpha_{jk} \lambda_j^k \leq \varepsilon$이면 $\lambda_j^* \leftarrow \lambda_j^k$로 놓고 (step 9)으로 간다.

2) $\|d^k\| < \delta$이며 $\sum_{j \in K} \alpha_{jk} \lambda_j^k > \varepsilon$이면 $\hat{\varepsilon} \leftarrow \mu \hat{\varepsilon}$로 놓고 (step 5)로 간다.

(step 7) 번들 수정

1) 현재 번들의 크기 |K| 가 b보다 작으면

$K \leftarrow \{j \in K \mid \lambda_j^k > 0\}$,

$D \leftarrow \{\pi^j \mid j \in K\}$, $(x_1^j, x_2^j, \cdots, x_N^j)$

$P \leftarrow \{(x_1^j, x_2^j, \cdots, x_N^j) \mid j \in K\}$로 수정한다.

2) 현재 번들의 크기 |K| 가 b일 때

$\pi^k \leftarrow d^k$,

$D \leftarrow \{\pi^k\}$,

$P \leftarrow \{(x_1^k, x_2^k, \cdots, x_N^k)$

$= \sum_{j \in K} \lambda_j^k (x_1^j, x_2^j, \cdots, x_N^j)\}$,

$a_{kk} \leftarrow \sum_{j \in K} \alpha_{jk} \lambda_j^k$,

$K \leftarrow \{k\}$로 수정한다.

(step 8) 번들-분해법의 선형탐색

1) 아래 조건 (a), (b), (c)를 만족하는 $u^{k+1} = y^k + t_k d^k$를 선택한다.

(a) $t_k > 0$

(b) $\langle \pi^{k+1}, d^k \rangle \geq \beta v_k$

(c) $g(u^{k+1}) \geq g(y^k)+rt_k v_k$ (serious step)이거나

$\alpha(y^k, u^{k+1}, \pi^{k+1}) \leq \mu\varepsilon_k$ (null step)이 되게 한다.

여기서 $\alpha(y^k, u^{k+1}, \pi^{k+1}) \equiv g(u^{k+1})-g(y^k) + \langle \pi^{k+1}, y^k-u^{k+1} \rangle$

$g(u^{k+1})$ 및 $\pi^{k+1} \in \partial g(u^{k+1})$은 $(QP(u_{k+1})_i)$를 풀어 얻는다.

2) serious step 이면

$y^{k+1} \leftarrow u^{k+1}$

$\alpha_{j,k+1} \leftarrow \alpha_{jk} + g(y^k)-g(y^{k+1})$
$+ \langle \pi^j, y^{k+1}-y^k \rangle, j \in K$

$\alpha_{k+1,k+1} \leftarrow 0$

$K \leftarrow K \cup \{k+1\}$

$D \leftarrow D \cup \{\pi^{k+1}\}$

$P \leftarrow P \cup \{(\pi_1^{k+1}, \pi_2^{k+1}, \cdots, \pi_N^{k+1})\}$로 놓는다.

3) null step 이면

$y^{k+1} \leftarrow y^k$

$\alpha_{j,k+1} \leftarrow \alpha_{jk}, j \in K$

$\alpha_{k+1,k+1} \leftarrow \alpha(y^{k+1}, \mu^{k+1}, \pi^{k+1})$

$K \leftarrow K \cup \{k+1\}$

$D \leftarrow D \cup \{\pi^{k+1}\}$

$P \leftarrow P \cup \{(x_1^{k+1}, x_2^{k+1}, \cdots, x_N^{k+1})\}$로 놓고 (step 5)로 간다.

(step 9) SQA 선형탐색

1) $\bar{x} = (\bar{x}_1, \bar{x}_2, \cdots, \bar{x}_N)$를 $\bar{x}_i = \sum_{j \in K} \lambda_j x_i^j$ (j=1, 2,…, N)로 놓는다.

2) 아래조건 (a), (b)를 만족하면서

$\tilde{x} = \hat{x} + \tilde{t}(\bar{x} - \hat{x})$이 되게 한다.

(a) $\tilde{t} = 1$ 즉 $\tilde{x} = \bar{x}$ (국소수렴을 적용할 때)

(b) $\tilde{t} = argmin \{f(\hat{x} + t(\bar{x} - \hat{x})) \mid 0 \leq t \leq 1\}$(전체수렴을 적용할 때)

(step 10) SQA 수렴성 조사

1) $\|f(\tilde{x}) - f(\hat{x})\| < \tau$ 이면 $x^* \leftarrow \tilde{x}$로 놓고 멈춘다.

2) $\|f(\tilde{x}) - f(\hat{x})\| > \tau$ 이면 $\hat{x} \leftarrow \tilde{x}$로 놓고 (step 1)으로 간다.

4. SQA 알고리즘의 수렴성

SQA 알고리즘의 수렴성은 내부 반복인 번들-분해법에 대한 수렴성과 외부 반복에 대한 SQA 수렴성으로 나누어 조사한다. 먼저 결합조건식이 부등식인 경우의 콘벡스 최적화 문제 (4. 1)에 대한 번들-분해법의 수렴성에 대해서 조사한다.

$$\text{Minimize } \sum_{i=1}^{n} f_i(x_i) \tag{4.1}$$

$$\text{Subject to } \sum_{i=1}^{n} A_i x_i \leq a$$

문제 (4. 1)의 쌍대문제는

$$\text{Minimize } g(y) \equiv -\langle y, a \rangle - \sum_{i=1}^{n} f_i^*(-A_i^T y)$$

$$\text{Subject to } y \geq 0 \tag{4.2}$$

이 된다. 문제 (4. 1)의 결합조건식이 부등식인 경우의 수렴성은 Robinson[12](정리 4. 1)의 결합조건식이 등식인 경우의 수렴성과 유사하게 얻어진다.

[정리 2] 함수 $f_i : R^m \rightarrow (-\infty, +\infty)$는 x_i에

대해 콘벡스(closed proper convex)이고 A_i는 $R^{n_i} \rightarrow R^m$의 선형변환이고 $a \in R^m$이다. 다음의 (i)-(ii)를 가정한다.

(i) R^m의 음의 벡터 d에 대해 다음 시스템의 해가 존재하고

$$\sum_{i=1} A_i x_i \leq a+d,\ x_i \in \text{dom } f_i\ (i=1,\cdots,n) \tag{4. 3}$$

(ii) 임의의 실수 r에 대해서 다음 집합이 제한(bounded)되어 있다.

$$\{(x_1, \cdots, x_n) \mid \sum_{i=1}^{n} A_i\ x_i \leq a,\ \sum_{i=1}^{n} f_i\ (x_i) \leq r\} \tag{4.4}$$

그러면 임의의 $\eta>0$에 대해서 다음의 (a)-(d)를 만족하는 $\hat{x}_1,\cdots,\hat{x}_n$, $\hat{y}$와 (e)-(f)를 만족하는 $\bar{x}_1, \cdots, \bar{x}_n$, $\bar{y}$에 대해 $\|\hat{x}-\bar{x}_n\| \langle\eta,\cdots,\|\hat{x}_n-\bar{x}_n\| \langle\eta,\|\hat{y}-\bar{y}\| \langle\eta$이 만족되는 $\delta>0$와 $\varepsilon>0$이 존재한다.

(a) 각 $i=1,\cdots,n$ 및 $j=1,\cdots,k$에 대해서 x_i^j가 $f_i(\cdot)-\langle -A_i^T y^j, \cdot\rangle$의 최소해이다. 즉, $-A_i^T y^j \in \partial f_i(x_i^j)$이 성립된다.

(b) $j=1,\cdots,k$에 대해서 $\pi^j = -a+\sum_{j=1}^{n} A_i x_i^j$ 라고 하면 $\pi^j \in \partial g_i(y^j)$이 성립된다.

(c) $d=\sum_{j=1}^{n} \lambda_j\ \pi^j$ 및 $\varepsilon_j = g(y^j)-g(y^k)-\langle y^j-y^k, \pi_j\rangle$라고 할 때, $\sum_{j=1}^{n} \lambda_j = 1$ $\|d\| \leq \delta$ 및 $\sum_{j=1}^{n} \lambda_j\ \varepsilon^j \leq \varepsilon$를 만족하는 비음의 $\lambda_1, \cdots,\lambda_k$가 존재한다.

(d) $\hat{y} = y^k$ 및 $\hat{x}=\sum_{j=1}^{n} \lambda_j\ x_i^j (i=1,\cdots,n)$이다.

(e) $(\bar{x}_1,\cdots,\bar{x}_n)$은 Minimize $\{\sum_{i=1}^{n} f_i(x_i) \mid \sum_{i=1}^{n} A_i\ x_i \leq a\}$의 최적해이다.

(f) $\bar{y}$ 는 Maximixe $\{g(y) \mid y\geq 0\}$의 최적해이다. ($g(y)=-\langle y, a\rangle - \sum_{i=1}^{n} f_i^*(-A_i^T y)$임)

(증명) 증명과정은 Robinsion[12]의 정리 4.1과 유사하다. 먼저 $x=(x_1,\cdots,x_n)\in R^N$ 및 $N=\sum_{i=1}^{n} n_i$이라고 정의하면 $f(x)=\sum_{i=1}^{n} f_i(x_i)$가 된다. 또한 $A=[A_1|A_2|\cdots|A_n]$라고 정의하면 $A:R^N \rightarrow R^m$인 선형변환이며 $Ax = \sum_{i=1}^{n} A_i\ x_i$이 된다. 인자 (ε, r, s)을 갖는 다가함수(multifunction) M을 다음과 같이 정의한다.

$$M(\varepsilon, r, s)=\{(x, y) \mid \binom{r}{s} \in \begin{pmatrix} \partial_\varepsilon f(x) & -A^T \\ A & 0 \end{pmatrix} \binom{x}{y} + \binom{0}{-a}\} \tag{4.5}$$

그러면 임의의 $s\leq 0$에 대한 $M(0,0,s)$는 문제 (4.1)의 원문제 및 쌍대문제의 해집합이 된다. 이제 번들-분해법의 수렴성은 다가함수 M이 임의의 $s\leq 0$에 대한 임의의 점 $(0,0,s)$에서 Hausdorff 상반연속(upper-semicontinous)임을 보이면 된다. 즉 ε가 r가 충분히 0에 접근할 때, $M(\varepsilon, r, s)$의 각 점이 $M(0, 0, s)$ 내의 어떤 한점으로 부터 미리 정한 거리 내에 놓이게 된다는 것을 보이면 된다. [12]의 정리 3. 4에 의하면 $\varepsilon\geq 0$ 가 주어질 때, 다음 두 조건을 만족하는 (r, s)에 대해 M은 $(\varepsilon, r, s) \in R_+ \times R^N \times R^m$에서 Hausdorff 상반연속이다.

$$r \in \text{int}[\text{dom } f^* + im\ A^T] \tag{4.6}$$

$$s \in \text{int}[A(\text{dom } f)-a] \tag{4.7}$$

조건 (4. 7)은 Slater 형태의 가정 (i)과 동일하고 조건 (4. 6)는 Robinson[12]의 4절에서 보인 바와 같이 compact-level-set 가정인 (ii)와 동일하다. ■

외부 반복인 SQA의 수렴성은 국소수렴(local convergence) 및 전체수렴(global convergence) 모두 비선형계획문제의 IQP(Iterative Quadratic Programming)법의 수렴정리[5,6]를 이용한다. IQP법의 수렴정리를 (LCOP)에 적용하기 위해 다음의 용어 및 표현을 정의한다.

- 블록-대각 행렬 $M(x)$: 문제 (LCOP)의 목적함수 $f(x)$의 Hessian 행렬 $\nabla^2 f(x)$를 SQA로 근사화시킨 블록-대각 행렬. 즉, $M(x)$의 i번째 대각 블록은 제 2 절의 식 (2. 4)의 M_i이다.
- Lagrangian 함수 $L(z)$: (4. 8)로 정의한다.

$$L(z) = L(x_N, \cdots, x_N, u_1, \cdots, u_N, y)$$
$$= f(x_1, \cdots, x_N)+\sum_{i=1}^{N} \langle u_i, B_i x_i - b_i \rangle$$
$$+ \langle y, \sum_{i=1}^{N} A_i x_i - a \rangle \qquad (4.8)$$

- 2차 충분조건 및 Strict complementarity: [4]를 참조할 수 있다.
- $\beta = \frac{3}{2} \| \nabla_z h(z)^{-1} \|$라고 놓는다.

$$h(z)= \begin{pmatrix} \nabla_{x_1} L(z) \\ \vdots \\ \nabla_{x_N} L(z) \\ Y\nabla_y L(z) \end{pmatrix} \text{ 이고}$$

$$Y = \begin{pmatrix} y_1 & 0 & \cdots & 0 \\ 0 & y_2 & \cdots & 0 \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ 0 & 0 & \cdots & y_m \end{pmatrix} \text{ 이다}$$

[정리 3] (국소수렴) ($\bar{x}$, 0, $\bar{y}$)가 문제 (LCOP)의 Kuhn-Tucker 점으로서 2차 충분조건, strict complementarity를 만족하며 $\bar{x}$에서 모든 등식으로 성립되는 제약식의 gradient 들이 일차독립이라고 가정한다. $\bar{x}$의 open neighborhood에서 f의 2차 도함수가 Lipschitz 연속이라고 가정한다. 또한

(i) 시스템 (4.9)의 해가 존재하고

$$\begin{cases} B_i x_i = b_i, \; i,\cdots,N \\ \sum_{i=1}^{N} A_i x_i < a \end{cases} \qquad (4.9)$$

(ii) 임의의 실수 r에 대해서 집합 (4.10)이 제한되어 있다.

$$\{(x_1, \cdots, x_N) \mid \sum_{i=1}^{N} A_i x_i \le a,$$
$$f(x_1, \cdots, x_N) \le r\} \qquad (4.10)$$

그러면 $\|(x^0, y^0)-(\bar{x}, \bar{y})\| < \delta_1$ 및 $\|M(x^0) - \nabla^2 f(\bar{x})\| < \delta_2$을 만족하는 임의의 시작점 (x^0, y^0)으로 부터 QSA 알고리즘에 의해 생성되는 개선점들의 수열 $\{(x^k, y^k)\}$이 존재하고 블록-대각 행렬 $M(x^k)$가 다음중 어느 하나 또는 그 이상을 만족하면

(a) $$\|M(x^k)-\nabla^2 f(x^k)\| \le \frac{1}{10\beta} \qquad (4.11)$$

(b) $$\begin{cases} \| M(x^k)-\nabla^2 f(x^k) \| \le \dfrac{1}{10\beta} \\ \left\| \dfrac{(M(x^k)-\nabla^2 f(x^k))(x^{k+1}-x^k)}{\|(x^{k+1}, y^{k+1})-(x^k, y^k)\|} \right\| \to 0 \end{cases} \qquad (4.12)$$

(c) $$M(x^k) = \nabla^2 f(x^k) \qquad (4.13)$$

$(\bar{x}, \bar{y})$에 수렴하며 수렴율(R-rates)은 각각 (a) R-linear 수렴율, (b) R-superlinear 수렴율 (c) R-quadratic 수렴율이 되는 양수 δ_1 및 δ_2가 존재한다.

(증명) SQA 근사화로 분할된 (2. 5)의 함수 f_i에 대해서 가정 (i) 및 (ii)는 각각 [정리 2]의 가정 (i) 및 (ii)를 만족한다. 따라서 QSA 알고리즘은 번들-분해법에 의해 개선점을 얻게 된다. 이후 QSA 알고리즘의 수렴성 및 수렴비율에 대한 증명은 [5]의 정리 3. 1에 의한다. 다만 각 반복에서 분리되는 제약등식의 승수 u^k

$= (u_1^k, \cdots, u_N^k)$는 0으로 놓는다. ■

[정리 4] (전체수렴) 함수 f의 2차 도함수는 연속이다. [정리 3]의 (i), (ii)와 함께 다음의 (iii)과 (iv)를 가정한다.

(iii) 각 반복 k에서 다음 부등식을 만족하는 $\beta \geq r > 0$ 가 존재한다.

$$r\|x\|^2 \leq \langle M(x^k)x, x\rangle \leq \beta \|x\|^2 \qquad (4.14)$$

(iv) 각 반복 k에서 계산되는 쌍대문제의 해 y^k는 일양제한(uniformly bounded)된다. 즉, 어떤 $r>0$가 존재해서 모든 k에 대해 $\|y^k\|_r \leq r$이 성립된다.

그러면 QSA 알고리즘에 의해 생성되는 수열 $\{(x^k, 0, y^k)\}$는 문제 (LCOP)의 Kuhn-Tucker 점에서 멈추거나 행렬 B_i (i=1, …, N)가 일차독립일 때에 임의의 accumulation point $(\bar{x}, 0, \bar{y})$는 Kuhn-Tucker 점이다.

(증명) [6]의 정리 3. 2를 적용한다. ■

5. SQA 알고리즘 적용

4절에서 구현한 SQA 알고리즘을 ANSI C 프로그램으로 작성하여 Sun sparc II에서 여러 형태의 문제에 적용하였다. 또한 SQA 알고리즘 적용 결과와 GAMS/MINOS 적용 결과를 비교하였다.

먼저 작은 규모의 비분리 2차계획 문제에 대해서 SQA 알고리즘을 적용한 문제 형태는 [문제 1]과 같이 목적함수의 행렬이 positive definite인 경우로서 제약식의 형태는 [문제 1]에서 제시한 형태를 이용하되 추가되는 비분리 목적함수를 다양하게 변화시켜 적용하였다.

[문제 1] 비분리 2차계획문제

$$\text{Min } \frac{1}{2}\left\langle \begin{pmatrix} 5 & -1 & 1 & 0 \\ -1 & 5 & 2 & 1 \\ 1 & 2 & 5 & 0 \\ 0 & 1 & 0 & 5 \end{pmatrix}\begin{pmatrix} x_{11} \\ x_{12} \\ x_{13} \\ x_{14} \end{pmatrix}, \begin{pmatrix} x_{11} \\ x_{12} \\ x_{13} \\ x_{14} \end{pmatrix}\right\rangle + \frac{1}{2}\left\langle \begin{pmatrix} 5 & -1 & 1 & 0 \\ 1 & 5 & 2 & 1 \\ -1 & 2 & 5 & -1 \\ 1 & 1 & -1 & 5 \end{pmatrix}\begin{pmatrix} x_{21} \\ x_{22} \\ x_{23} \\ x_{24} \end{pmatrix}, \begin{pmatrix} x_{21} \\ x_{22} \\ x_{23} \\ x_{24} \end{pmatrix}\right\rangle$$

$$+ x_{11} + 2x_{12} - 2x_{13} + x_{14} - x_{21} + 2x_{22} + 3x_{13} - 5x_{14}$$

+ {추가되는 비분리 목적함수식}

제약조건 :

$$\begin{array}{lr} x_{14} - x_{12} + 3x_{13} + 4x_{14} & (\geq \text{ or } =)\ 5 \\ 5x_{11} + 2x_{12} \qquad + 2x_{14} & (\geq \text{ or } =)\ 14 \\ \qquad 3x_{21} \qquad + x_{23} + 2x_{24} & (\geq \text{ or } =)\ 6 \\ \qquad -2x_{21} + x_{22} + x_{23} + 3x_{23} & (\geq \text{ or } =)\ 12 \\ x_{11} + 2x_{12} + x_{13} + 5x_{14} + 2x_{21} - x_{22} + 4x_{23} + x_{24} & [\leq \text{ or } =]\ 18 \\ 3x_{11} + x_{12} + 2x_{13} - x_{14} + 3x_{21} + 2x_{22} + x_{23} + 5x_{24} & [\leq \text{ or } =]\ 32 \end{array}$$

$$x_{ij} \geq 0 \ ; \ i=1, 2 \ \& \ j=1, \cdots, 4$$

〈표 5-1〉은 2차계획문제의 분리 제약조건식은 등식(또는 부등식)이고 결합조건식이 부등식(또는 등식)인 경우로 나누어서, 10가지 경우의 추가 비분리 목적함수식에 따른 SQA 알고리즘 적용결과와 GAMS/MINOS의 최적해를 비교한 것이다. 결과로부터 알수 있듯이 SQA 알고리즘은 GAMS/MINOS와 같은 최적해를 구하였다.

〈표 5-1〉 SQA 알고리즘 적용결과 : 비분리 2차계획문제

번호	추가 비분리 목적함수	분리 제약식	결합 제약식	SQA 반복수	SQA 목적함수값	GAMS 목적함수값
	비분리 목적함수식이 없는 경우	≥	=	2	46.313332	72.776
		=	≤	3	63.454315	63.454
1	$13x_{13}x_{22}$	≥	=	3	46.313332	46.313
		=	≤	3	63.454315	63.454
2	$8x_{13}x_{23}$	≥	=	4	49.113220	49.113
		=	≤	3	63.454315	63.454
3	$15x_{12}x_{24}$	≥	=	4	48.354317	48.354
		=	≤	3	65.427475	65.427
4	$13x_{11}x_{24}+4x_{14}x_{21}$	≥	=	4	47.311111	47.311
		=	≤	3	63.454315	63.454
5	$4x_{14}x_{21}+10x_{14}x_{24}$	≥	=	4	49.271313	49.271
		=	≤	3	68.647865	68.648
6	$3x_{11}x_{24}+4x_{14}x_{21}$	≥	=	4	73.602753	73.603
		=	≤	3	81.81701	81.817
7	$15x_{12}x_{24}+8x_{13}x_{23}+4x_{14}x_{21}$	≥	=	3	52.573071	52.573
		=	≤	3	65.427475	65.427
8	$13x_{13}x_{22}+8x_{13}x_{23}+7x_{14}x_{23}$	≥	=	6	55.311600	55.312
		=	≤	5	63.454315	63.454
9	$15x_{12}x_{24}+13x_{13}x_{22}+8x_{13}x_{23}+4x_{14}x_{21}$	≥	=	3	52.573063	52.573
		=	≤	3	65.427460	65.427
10	$15x_{12}x_{24}+13x_{13}x_{22}+8x_{13}x_{23}+4x_{14}x_{21}+5x_{14}x_{22}$	≥	=	3	52.573063	52.573
		=	≤	3	74.594131	74.594

다음은 목적함수의 형태가 다양한 경우에 대해서 적용한 것으로 [문제 2]는 목적함수가 비분리 4차 다항식을 갖는 수리 계획문제이고 [문제 3]은 비분리 지수함수를 갖는 수리계획 문제이고, [문제 4]는 비분리 분수식을 갖는 수리계획문제로서 제약식은 모두 [문제 1]의 경우와 같다.

[문제 2] Min $2x_{11}^4 + x_{12}^4 + 3x_{13}^4 + 4x_{14}^4 + x_{21}^4 + 3x_{22}^4 + 2x_{23}^4 + 4x_{24}^4 + 4x_{11}^2 x_{23}^2 + 8x_{14}^2 x_{24}^2$

[문제 1]의 제약식

[문제 3] Min $e^{x_{11}} + e^{4x_{12}} + e^{x_{13}} + 4e^{x_{14}} + e^{x_{21}} + 7e^{3x_{22}} + e^{x_{23}} + e^{x_{24}} + e^{x_{11}+x_{23}} + e^{-x_{14}-x_{24}}$

[문제 1]의 제약식

[문제 4] Min $\dfrac{x_{11}+x_{21}}{10-x_{11}-x_{21}} + \dfrac{x_{12}+x_{22}}{15-x_{12}-x_{22}} + \dfrac{x_{13}+x_{23}}{15-x_{13}-x_{23}} + \dfrac{x_{14}+x_{24}}{10-x_{14}-x_{24}}$

[문제 1]의 제약식

〈표 5-2〉에는 [문제 2], [문제 3] 및 [문제 4]에 대한 SQA 알고리즘 적용 결과와 GAMS/MINOS의 결과가 요약되어 있다. 특히, [문제 2]의 경우에 목적함수를 2차 근사화하면 Hessian 행렬이 positive semi-definite가 되지 않아 분리제약식이 부등호이고 결합 제약식이 등호인 경우에 대해서는 결과를 얻지 못한다. 그러나 다른 경우에 대해서는 정확한 해를 구하였다.

〈표 5-2〉 SQA 알고리즘 적용결과: 비분리 수리계획문제

문 제	분리 제약식	결합 제약식	SQA iteration수	SQA 최적 목적함수값	GAMS/MINOS 목적함수값	비 고
[문제 2]	=	≤	5	612.069335	612.069	
	≥	=	–	–	176.800	(주1)
[문제 3]	=	≥	7	56763.399402	56763.399	
	=	≤	7	28.04437	28.020	
[문제 4]	=	≤	4	1.118862	1.119	
	=	≤	9	1.296039	1.296	

(주1) Hessian 행렬이 positive semi-definite가 안됨

이제 SQA 알고리즘의 효율성을 조사하기 위해 대규모 비분리 2차계획 문제에 적용하였다. [문제 5]는 대규모 비분리 2차계획 문제로서 각 목적함수 계수, 블록-삼각 구조의 제약식 및 우변의 계수들을 ANSI C의 랜덤함수를 이용하여 생성하였다.

[문제 5] 대규모 비분리 2차계획문제

$$\text{Min } \frac{1}{2}\langle M(x_1, x_2, \cdots, x_n), (x_1, x_2, \cdots, x_n)\rangle + \sum_{i=1}^{N} c_i x_i$$

제약조건

$$\begin{aligned} B_1 x_1 \quad\quad\quad\quad &= b_1 \\ B_2 x_2 \quad\quad &= b_2 \\ \cdot \quad &= \cdot \\ B_N x_N &= b_N \\ A_1 x_1 + A_2 x_2 + \cdots + A_N x_N &\le a \\ x_1 \ge 0,\ x_2 \ge 0,\ \cdots,\ x_N \ge 0 & \end{aligned}$$

각 계수를 얻은 방법은 다음과 같다. rand()는 [0,1]사이의 난수값을 발생시켜주며 일양분포를 따르는 랜덤함수이다. 먼저 가능해를 생성한다.

(1) 가능해 생성: $(x_i)_j$, $i=1,\cdots,N$, $j=1,\cdots,m_i$

목적함수의 계수는 다음과 같은 절차에 따라 구하였다.

(2) 상삼각(upper triangular) 행렬 W를 만든다.

$$\begin{cases} W_{ij} = 10 \times \text{rand}(\ ) \\ W_{ij} = 10 \times \text{rand}(\ ) - 1.0) \end{cases}$$

$$i = 1, \cdots, N,\ j = 1, \cdots, m_i\ (i<j).$$

(3) $M = W^T W$을 계산하면 M는 positive semi-definite가 되며 만일 W가 nonsingular 행렬이면 M은 positive definite가 된다.

블록-삼각 구조의 제약조건식의 계수는 다음 절차에 따라서 구한다.

(4) 분리제약식의 계수: $B_i x_i = b_i$

에서 $(B_i)_{jk}$는 랜덤함수를 이용하여 얻고 우변 b_i의 값을 다음과 같이 구한다.

$$\begin{cases} (B_i)_{jk} = rand(\ ) \times 10 \\ (b_i)_j = \sum_{k=1}^{ni} (B_i)_{jk}\ (x_i)_k \end{cases}$$

$$i = 1, \cdots, N,\ j = 1, \cdots, m_i,\ k = 1, \cdots, n_i.$$

(5) 결합제약식의 계수 : 먼저 A_i의 계수들을 얻은 후에 $\sum_{i=1}^{N} A_i x_i$을 계산하여 부등식이 되도록 양수의 랜덤값을 우변에 더한다.

$$\begin{cases} (A_i)_{jk} = rand(\ ) \times 10 \\ a = \sum_{i=1}^{N} A_i x_i + rand(\ ) \times 10 \end{cases}$$

$$i = 1, \cdots, N,\ j = 1, \cdots, m_i,\ k = 1, \cdots, n_i.$$

위의 절차에 의한 다양한 크기의 분리 제약식과 결합제약식을 가지는 대규모 비분리 2차계획문제 조합은 〈표 5-3〉과 같다. 〈표 5-3〉에서 하부문제 수는 삼각-블록구조 제약식에서 블록의 수(분리 제약식 수)에 해당하며 하부문제의 크기는 각 블록의 크기에 해당한다. 또한 밀도는 전체문제의 행렬의 차원을 각 블록들과 결합 제약식들에 의해 만들어지는 행렬 요소의 숫자로 나눈 값이다.

〈표 5-3〉의 문제에 각각 SQA 알고리즘을 적용하였다. 수렴성, 반복회수 및 수행시간 등의 결과는 〈표 5-4〉와 같다. 여기서 X-norm은 수

〈표 5-3〉 대규모 비분리 2차계획문제 형태

문제번호	전체문제 크기	결합제약식수	하부문제수	하부문제 크기	밀도(%)
1	123 × 200	3	40	3 × 5	4.88
2	103 × 200	3	20	5 × 10	7.77
3	54 × 200	4	10	5 × 20	16.67
4	203 × 300	3	50	4 × 6	3.45
5	214 × 300	4	30	7 × 10	5.14
6	148 × 300	8	20	7 × 15	10.14
7	256 × 400	6	50	5 × 8	3.71
8	286 × 400	6	40	7 × 10	4.55
9	148 × 400	8	20	7 × 20	10.14

렴성의 정도($\|x^{k+1}-x^k\|$)를 나타내며 SQA 회수는 SQA 알고리즘의 반복수이고 BBD 회수는 SQA 알고리즘이 끝날 때까지 적용된 번들-분해법의 내부 반복수의 합계이다. 문제 7 ~ 9의 경우는 GAMS 수행시 'Ran Out of Memory'로 인해 결과를 얻지 못하였다. 〈표 5-4〉의 결과로부터 GAMS/MINOS 에 비해 SQA 알고리즘이 블록-삼각의 선형 제약식의 구조를 갖는 대규모 비분리 콘벡스 2차계획 문제의 해를 수배 이상의 빠른 시간으로 효과적으로 발견한다고 하겠다.

본 연구의 동기가 된 광대역 종합정보통신망의 ATM망 가상경로 재구성 모형에의 SQA 적용 결과는 별도의 논문[2]을 참고할 수 있다.

6. 결 론

본 연구에서는 선형의 블록-삼각구조를 갖는 대규모 비분리 콘벡스 최적화 문제의 해법을 제시하였다. 이와같은 형태의 문제는 생산일정, 재고관리 및 분배계획 문제, 산업간 입력-출력 분석, multicommodity network flow 문제 등 많은 응용이 있으나 본 연구의 동기는 광대역 종합정보통신망의 ATM망 가상경로 재구성 연구에서 비롯되었다. [1]

본 연구에서는 목적함수가 미분가능할 때 축차적으로 목적함수를 2차함수로 근사화시켜 번들-분해법을 적용하는 SQA 알고리즘을 제시하였다. 내부 반복에 해당되는 분리가능한 2차함수의 목적함수를 갖는 블록-삼각구조의 대규모 콘벡스 최적화 문제을 푸는 번들-분해법을 구현하였다. 이는 Medhi[10]가 분리가능한 선형 함수에 대해 번들-분해법을 구현한 연구의 확장연구에 해당된다.

목적함수가 분리되지 않는 경우는 목적함수

〈표 5-4〉 대규모 비분리 2차계획문제에 대한 SQA 알고리즘 적용 결과

문제 번호	X-norm	BBD 회수	SQA 회수	평균 BBD 회수	OBJ 값		Execution-time(sec)	
					SQA	GAMS/MINOS	SQA	GAMS/MINOS
1	0.000034	15	4	3.75	113615.168643	113615.17	35	146
2	0.000054	48	3	16	77112.819821	77112.82	68	349
3	0.000013	39	3	13	40832.547822	40832.55	63	348
4	0.000025	45	4	11.25	182615.367432	182615.37	66	833
5	0.000049	31	4	7.75	199581.664704	199581.66	107	835
6	0.000075	30	4	7.5	103695.542871	103695.54	91	826
7	0.000008	36	5	7.6	201321.188373	-	87	-
8	0.000012	40	4	10	687203.381262	-	110	-
9	0.000081	33	5	6	573633.839211	-	130	-

주) 평균 BBD 회수 = (BBD 회수/SQA 회수) 임

를 2차함수로 근사화시켜 번들-분해법을 축차적으로 적용하는 SQA 알고리즘을 구현하고 수렴성을 검토하였으며 기술적인 조건하에서 국소수렴 및 전체수렴 모두 가능하다. 또한 다양한 형태, 다양한 크기의 문제에 대해서 적용하여 보았다. 적용결과는 SQA법이 GAMS/MINOS 보다 수배 이상 빠른 시간내에 대규모 비분리 콘벡스 최적화문제를 효율적으로 해결하였다.

참 고 문 헌

[1] 박구현, "ATM 망의 가상경로 루팅 최적화", 한국경영과학회지, 제 20 권, 제 1 호, 1995.

[2] 박구현, 신용식, "ATM 망에서의 가상경로 설계", 한국통신학회논문지, 게재확정, 1996.

[3] Ahuja,R. K., Magnanti,T. L. & Orlin,J. B., *Network Flows: Theory, Algorithms, and Applications*, Prentice-Hall, 1993.

[4] Bazaraa,M. & Shetty,C. M., *Nonlinear Programming: Theory and Algorithms*, John Wiley and Sons Inc., 1979.

[5] Garcia Palomares,U. M. & Mangasarian, O. L., "Superlinearly Convergent Quasi-Newton Algorithms for Nonlinearly Constrained Optimization Problems", *Mathematical Programming*, Vol. 11, 1976.

[6] Han,S. P., "A Globally Convergent

Method for Nonlinear Programming", *Journal of Optimization Theory and Applications*, Vol. 22, 1977. (Lecture Notes by O. L. Mangasarian)

[7] Lasdon,L. S., Optimization Theory for Large Systems, Macmillan, 1970.

[8] Lemarechal,C., "Bundle Methods in Nonsmooth Optimization", Nonsmooth Optimization, Ed. by C. Lemarechal & R. Mifflin, IIASA V3, Pergamon, 1978.

[9] Lemarechal,C., Strodiot,J. J. & A., "On a Bundle Algorithm for Nonsmooth Optimization", Nonlinear Programming 4, edited by O. L. Magasarian, R. R. Meyer & S. M. Robinson, Academic Press, 1981.

[10] Medhi,D., "Bundle-based Decomposition for Large-scale convex Optimization: Error estimate and Application to Block-angular Linear Programs", Mathematical Programming, Vol 66, 1994.

[11] Robinson,S. M. "Bundle-based Decomposition: Description and Preliminary Results", in A. Prekopa, J. Szelezsan, and B. Strazicky, eds., System Modelling and Optimization, Springer-Verlag, 1986.

[12] Robinson,S. M. "Bundle-based Decomposition: Conditions for Convergence", Annales de l'Institute Henri Poincare: Analyse Non Lineaire 6, 1989. (Technical Report, Department of Industrial Engineering, University of Wisconsin-Madison, 1989.)

[13] Rockafellar,R. T., Convex Analysis, Preceton Press, 1970.

[14] Rockafellar,R. T., Conjugate Duality and Optimization, SIAM, Philadelphia, 1974.