On the Solution to the Nonlinear Resource Allocation Problem Using Variable Fixing and Interior Point Methods

James Rohal
rohaljj@muohio.edu

Miami University

May 29, 2009

Presented in fulfilment for the Masters in Science in Mathematics at Miami University in Oxford, OH under the supervision of Dr. Stephen Wright (wrightse@muohio.edu).

Outline

1. Resource Allocation Problem
2. Structural Properties & Earlier Algorithms
3. Interior Point Methods
4. Indicators
5. Numerical Tests
6. Future Work
7. Bibliography

Nonlinear Resource Allocation Problem

$\boldsymbol{\xi} \in \mathbb{R}^n$ is our decision variable.

$$
\begin{aligned}
(P) \qquad \min \quad & f(\boldsymbol{\xi}) := \sum f_i(\xi_i) \\
\text{st} \quad & g(\boldsymbol{\xi}) := \sum g_i(\xi_i) \leq M \\
& \ell_i \leq \xi_i \leq u_i,
\end{aligned}
$$

$f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^n$ are separable, convex, and differentiable functions. We assume WLOG that f_i and g_i are monotone.

Nonlinear Resource Allocation Problem

We expect the constraint to be active at an optimal solution.

$$
\begin{aligned}
(P) \qquad \min \quad & \sum f_i(\xi_i) \\
\text{st} \quad & \sum g_i(\xi_i) = M \\
& \ell_i \leq \xi_i \leq u_i .
\end{aligned}
$$

Stratified Sampling

Estimating a population mean with a value by minimizing the variance of the estimate subject to a linear sampling budget constraint.

$$
\begin{aligned}
\text{(SAMP)} \qquad \min \quad & \sum \frac{d_i}{\xi_i} - \frac{d_i}{N_i} \\
\text{st} \quad & \sum b_i \xi_i \leq M \\
& \ell_i \leq \xi_i \leq u_i \\
& \xi_i \text{ is integral.}
\end{aligned}
$$

Quadratic Knapsack

Minimize a quadratic objective function subject to a linear capacity constraint.

$$
\begin{aligned}
\text{(QP)} \qquad \min \quad & \sum \left(\frac{1}{2} p_i \xi_i^2 - a_i \xi_i \right) \\
\text{st} \quad & \sum b_i \xi_i \le M \\
& \ell_i \le \xi_i \le u_i \\
& \xi_i \text{ is integral.}
\end{aligned}
$$

Manufacturing Capacity Planning

Minimum cost selection of the service rate (or capacity) at each work station subject to an upper limit on the total dollar value of work-in-process in the system.

$$
\begin{aligned}
\text{(MCP)} \qquad \min \quad & \sum c_i \xi_i \\
\text{st} \quad & \sum b_i \left(\frac{\alpha_i}{\xi_i - \alpha_i} \right) \leq M \\
& \ell_i \leq \xi_i \leq u_i \\
& \xi_i \text{ is integral.}
\end{aligned}
$$

Previous Work by Others

- A survey of the continuous nonlinear resource allocation problem by Michael Patriksson [4].
  - Lagrange multiplier methods.
  - Pegging methods.
- Kiwiel [3] speeds up pegging algorithm of Bretthauer & Shetty [1].

Optimality Conditions

An optimal solution $\boldsymbol{\xi}^*$ with corresponding multiplier ρ^*, satisfies the KKT conditions:

- Lagrange Multiplier Equation: $\nabla f\left(\boldsymbol{\xi}^*\right) + \rho^* \nabla g\left(\boldsymbol{\xi}^*\right) = 0$.
- Primal Feasibility: $g\left(\boldsymbol{\xi}^*\right) - M \leq 0 \quad \text{and} \quad \ell_i \leq \xi_i^* \leq u_i$.
- Dual Feasibility: $\rho^* \geq 0$.
- Complementary Slackness: $\rho^*\left(g\left(\boldsymbol{\xi}^*\right) - M\right) = 0$.
- Tangent Criterion:

$$
\begin{aligned}
\xi_i^* = \ell_i, &\quad \text{if } f_i'\left(\xi_i^*\right) \geq -\rho^* g_i'\left(\xi_i^*\right) \\
\xi_i^* = u_i, &\quad \text{if } f_i'\left(\xi_i^*\right) \leq -\rho^* g_i'\left(\xi_i^*\right) \\
\ell_i < \xi_i^* < u_i, &\quad \text{if } f_i'\left(\xi_i^*\right) = -\rho^* g_i'\left(\xi_i^*\right).
\end{aligned}
$$

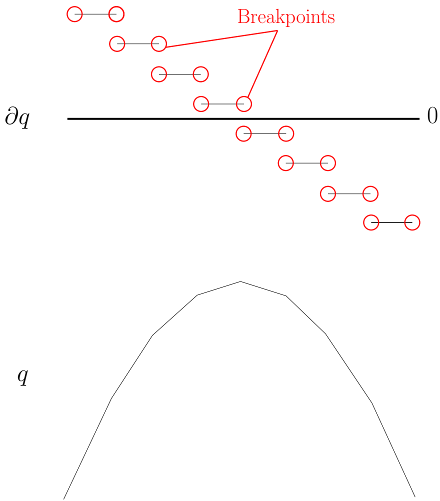
Lagrange Multiplier Algorithms

- Maximize the Lagrangian dual function (for given $\rho \geq 0$),

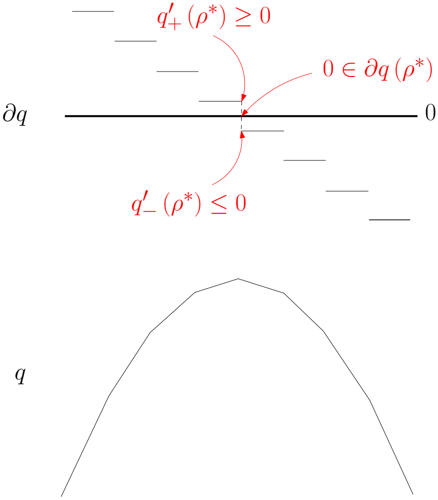
$$q(\rho) := -M\rho + \sum \min_{\ell_i \leq \xi_i \leq u_i} \{ f_i(\xi_i) + \rho g_i(\xi_i) \},$$

  i.e., maximize a concave piecewise linear function of ρ. We can find ρ^* using
  - Bisection search (iterative).
  - Breakpoint search.

Breakpoint Search



Breakpoint Search



Pegging Algorithms

Definition

A variable ξ_i is pegged if it is fixed at its upper bound u_i or lower bound ℓ_i.

- Solve relaxations of (P) where the bound constraints are relaxed.
- At each iteration at least one variable is fixed, so complexity is at most $O(n^2)$.
- Efficient algorithm if each subproblem can be solved in terms of the multiplier.

Pegging Framework

1. (Initialization) $L^k = \emptyset$, $U^k = \emptyset$.

For $k = 1$ **to** n **do**:

2. (Solve the subproblem). Let ξ^k denote the optimal solution (we assume one exists) to:

$$(P^k) \qquad \min \sum f_i(\xi_i)$$
$$\text{st} \sum g_i(\xi_i) = M, \quad \text{where } \xi_i = \begin{cases} u_i & \text{if } i \in U_k \\ \ell_i & \text{if } i \in L_k. \end{cases}$$

3. (Identify variables in (P^k) that do not satisfy their bounds)

$$\text{small}^k = \left\{i : \xi_i^k < \ell_i\right\} \quad \text{and} \quad \text{big}^k = \left\{i : u_i < \xi_i^k\right\}$$

If $\text{small}^k \cup \text{big}^k = \emptyset$, then an optimal solution has been found.

Pegging Framework (Continued)

4. (Calculuate total lower and upper infeasibilities).

$$\nabla = \sum_{i \in \mathrm{small}^k} \left[g_i(\ell_i) - g_i\left(\xi_i^k\right) \right] \qquad \Delta = \sum_{i \in \mathrm{big}^k} \left[g_i\left(\xi_i^k\right) - g_i(u_i) \right].$$

5. (Pegging) **If** $\nabla \geq \Delta$ then fix variables in big_k to upper bounds **Else** fix variables in small_k to lower bounds. Update U^{k+1}, and L^{k+1} appropriately. **Goto** step 2.

Interior Point Methods

- Travel through the interior to find a solution.
- A “barrier” or “penalty” function prevents the algorithm from approaching the boundary.
- Can use a primal-dual path following algorithm: iterates follow an arc of strictly feasible points (by staying within a neighborhood of it).
- Difficulties
  - Finding a good intial solution to start our algorithm.
  - Fast convergence requires problem specific tuning.

Forming a Nonnegative System (Restating the Optimality Conditions)

Introducing slacks x and s for the lower bound and upper bounds, respectively, forms a nonnegative system. We may assume through preprocessing that our constraint holds as equality.

$$
\begin{aligned}
(P') \quad \min \quad & \sum f_i(x_i + \ell_i) \\
\text{st} \quad & \sum g_i(x_i + \ell_i) = M \\
& u_i - \ell_i = s_i + x_i \\
& x_i \geq 0, s_i \geq 0.
\end{aligned}
$$

Optimality Conditions

The KKT conditions for the nonnegative system (P'). The variables $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ are the Lagrange multipliers for the lower bound and upper bound, respectively.

- Lagrange Multiplier Equation:

$$\nabla f(\boldsymbol{x}+\boldsymbol{\ell}) + \rho\nabla g(\boldsymbol{x}+\boldsymbol{\ell}) - \boldsymbol{\lambda} + \boldsymbol{\mu} = 0.$$

- Primal Feasibility:

$$\sum g_i(x_i+\ell_i) = M \qquad \text{and} \qquad u_i - \ell_i = s_i + x_i.$$

- Complementary Slackness: $\lambda_i x_i = 0$ and $\mu_i s_i = 0$.
- Nonnegativity: $x_i \geq 0$, $s_i \geq 0$, $\mu_i \geq 0$, $\lambda_i \geq 0$, $\rho \geq 0$.

Optimality Conditions and Central Path

The KKT conditions for the nonnegative system (P'). The variables $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ are the Lagrange multipliers for the lower bound and upper bound, respectively.

- Lagrange Multiplier Equation:

$$\nabla f(\boldsymbol{x}+\boldsymbol{\ell})+\rho\nabla g(\boldsymbol{x}+\boldsymbol{\ell})-\boldsymbol{\lambda}+\boldsymbol{\mu}=0.$$

- Primal Feasibility:

$$\sum g_i(x_i+\ell_i)=M \qquad \text{and} \qquad u_i-\ell_i=s_i+x_i.$$

- Complementary Slackness: $\lambda_i x_i=\tau$ and $\mu_i s_i=\tau$.
- Nonnegativity: $x_i>0$, $s_i>0$, $\mu_i>0$, $\lambda_i>0$, $\rho>0$.

Central Path

Definition

The central path is an arc of strictly feasible points where the complementary slackness conditions have the same value (τ) for all indices.

- Guides iterative procedure along a path avoiding spurious solutions.
- Reduces complementary slackness condition to zero at a steady rate.

IPM Framework

Let $\boldsymbol{z}^k = \left(\boldsymbol{x}^k, \boldsymbol{\lambda}^k, \boldsymbol{s}^k, \boldsymbol{\mu}^k, \rho^k\right)$.

1. Find a strictly feasible point $\boldsymbol{z}^0$.

For $k = 1, 2, \ldots$ do the following.

2. Solve the Newton system $JF\left(\boldsymbol{z}^k\right)\Delta\boldsymbol{z}^k = -F\left(\boldsymbol{z}^k\right)$ where

$$F\left(\boldsymbol{z}\right) = F(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{s}, \boldsymbol{\mu}, \rho) := \begin{bmatrix} \nabla f(\boldsymbol{x}+\boldsymbol{\ell}) + \rho\nabla g(\boldsymbol{x}+\boldsymbol{\ell}) - \boldsymbol{\lambda} + \boldsymbol{\mu} \\ \boldsymbol{\lambda}\cdot\boldsymbol{x} - \tau \\ \boldsymbol{\mu}\cdot\boldsymbol{s} - \tau \\ \boldsymbol{x} + \boldsymbol{s} + \boldsymbol{\ell} - \boldsymbol{u} \\ \sum g_i(x_i + \ell_i) - M \end{bmatrix}.$$

3. Set $\boldsymbol{z}^{k+1} = \boldsymbol{z}^k + \alpha^k\Delta\boldsymbol{z}^k$ where α^k is chosen so $\boldsymbol{z}^{k+1} > 0$.

Newton System

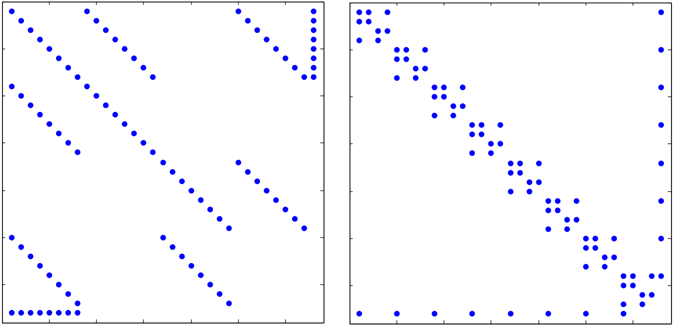
$$JF(\boldsymbol{z}) = JF(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{s}, \boldsymbol{\mu}, \rho) =$$

$$\begin{bmatrix} \operatorname{diag}\left(\frac{\partial^2 f}{\partial \xi_i^2} + \rho_i \frac{\partial^2 g}{\partial \xi_i^2}\right) & -I & 0 & I & \left(\frac{\partial g}{\partial \xi_i}\right)^T \\ \operatorname{diag}(\lambda_i) & \operatorname{diag}(x_i) & 0 & 0 & 0 \\ 0 & 0 & \operatorname{diag}(\mu_i) & \operatorname{diag}(s_i) & 0 \\ I & 0 & I & 0 & 0 \\ \left(\frac{\partial g}{\partial \xi_i}\right) & 0 & 0 & 0 & 0 \end{bmatrix}$$

- A pure Newton step has a tendency to take us to a point where the positivity condition $(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{s}, \boldsymbol{\mu}, \rho) > 0$ fails to hold.
- Counteract this by introducing a centering parameter $\sigma \in [0, 1]$ and a simple duality measure β. Replace τ by the product $\sigma\beta$.

Solving the Newton System

- Partition the Jacobian into blocks and find the inverse using the Schur complement.
- Gives a nice block structure.



Refinements

- Reduce dimension of the problem.
- Pegging implies there is a natural means for identifying active constraints.

Indicators

Recall $\boldsymbol{z}^k = (\boldsymbol{x}^k, \boldsymbol{\lambda}^k, \boldsymbol{s}^k, \boldsymbol{\mu}^k, \rho^k)$.

Definition

An indicator is a function I that identifies constraints that are active at a solution of a constrained optimization problem.

An indicator function I assigns to $\boldsymbol{z}^k$ an n-vector of extended reals and satisfies the property that if $\boldsymbol{z}^k \to \boldsymbol{z}^*$, then $\forall i$

$$\lim_{k\to\infty} I_i\left(\boldsymbol{z}^k\right) = \begin{cases} \theta_i & \text{if } i \text{ is inactive} \\ 0 & \text{if } i \text{ is active,} \end{cases}$$

for some θ_i satisfying $\min_i \theta_i > 0$.

Ideal Properties of an Indicator Function

1. Sharp separation property:

$$\min_{\text{inactive } i} \theta_i >> 0.$$

2. Uniform separation property:

$$\theta_i = \theta, \forall \text{ inactive } i,$$

   for some nonzero constant θ.

3. Inexpensive to compute.
4. $\{I(z^k)\}$ converges faster to its limit than $\{z^k\}$ converges to z^*.
5. Reliable early on in the iterative process.

Indicator Functions

- Variables as indicators.
  - Do not satisfy sharp or uniform separation.
  - Useful information is not given soon enough.
- Tapia indicator: Use quotient of successive Lagrange multipliers and the quotient of successive slack variables.

Tapia Indicator

- Lower bound indicator:

$$I_i^{\ell}\left(\boldsymbol{z}^k\right) = \frac{\boldsymbol{x}^{k+1}}{\boldsymbol{x}^k} + \left(1 - \frac{\boldsymbol{\lambda}^{k+1}}{\boldsymbol{\lambda}^k}\right).$$

- Upper bound indicator:

$$I_i^{u}\left(\boldsymbol{z}^k\right) = \frac{\boldsymbol{s}^{k+1}}{\boldsymbol{s}^k} + \left(1 - \frac{\boldsymbol{\mu}^{k+1}}{\boldsymbol{\mu}^k}\right).$$

- Both satisfy

$$\lim_{k\to\infty} I_i\left(\boldsymbol{z}^k\right) = \begin{cases} 2, & \text{if } i \text{ is inactive} \\ 0, & \text{if } i \text{ is active.} \end{cases}$$

Algorithms

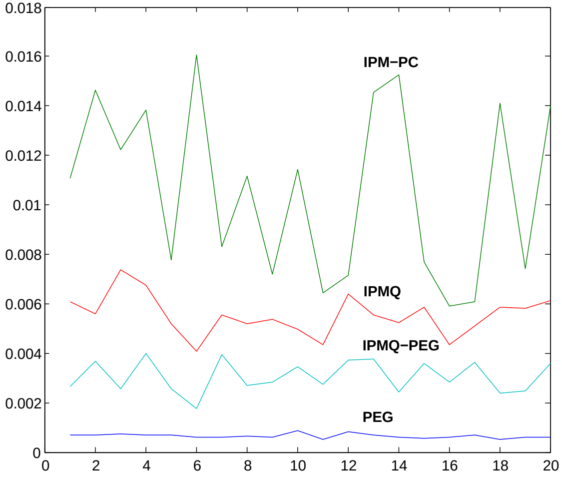
- **PEG**: Pegging method based on work by Bretthauer & Shetty [1] and later improved by Kiwiel [3].
- **IPMPC** vs. **IPMQ**: Two types of interior point methods.
  - **IPMPC**: Linearly convergent predictor-corrector method.
  - **IPMQ**: Quadratically convergent method based on work by Sun & Zhao [5].
- **IPMQ-PEG**: Algorithm using **IPMQ** above with pegging as subroutine.

Small Problems ($n = 40$)

Run	**PEG** (s)	**IPMQ-PEG** (s)
1	0.0007	0.0026
2	0.0007	0.0037
3	0.0007	0.0025
⋮	⋮	⋮
18	0.0005	0.0024
19	0.0006	0.0025
20	0.0006	0.0036

Run	**IPM-PC** (s)	**IPMQ** (s)
1	0.0111	0.0061
2	0.0146	0.0056
3	0.0122	0.0074
⋮	⋮	⋮
18	0.0141	0.0059
19	0.0074	0.0058
20	0.0140	0.0061

Subproblem with closed-form solution.

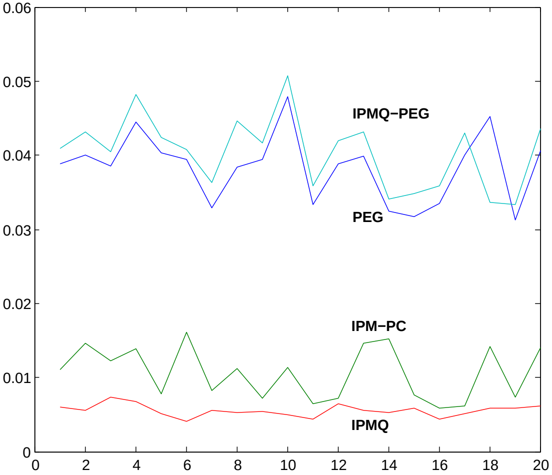


Small Problems ($n = 40$)

Run	**PEG** (s)	**IPMQ-PEG** (s)
1	0.0388	0.0409
2	0.0400	0.0431
3	0.0385	0.0404
⋮	⋮	⋮
18	0.0453	0.0337
19	0.0313	0.0333
20	0.0406	0.0436

Run	**IPM-PC** (s)	**IPMQ** (s)
1	0.0111	0.0061
2	0.0146	0.0056
3	0.0122	0.0074
⋮	⋮	⋮
18	0.0141	0.0059
19	0.0074	0.0058
20	0.0140	0.0061

Subproblem with numerical solution.



Large Problems

- Solution found in same number of iterations.
- **PEG**, **IPMQ**, **IPMQ-PEG** scale linearly with n.

Future Work

- Compare to breakpoint search method.
- Micro-optimizations.
- Consider applications without closed-form subproblem solutions.
- Indicators.
  - Identifying active variables sooner (less pegging iterations).
  - Tapia-Zhang.

Bibliography

[1] Bretthauer K, Shetty B.
The Nonlinear Resource Allocation Problem. (1995).

[2] El-Bakry AS, Tapia RA, Zhang Y.
A study of indicators for identifying zero variables in interior-point methods. (1994).

[3] Kiwiel K.
Variable Fixing Algorithms for the Continuous Quadratic Knapsack Problem. (2008).

[4] Patriksson M.
A survey on the continuous nonlinear resource allocation problem. (2008).

[5] Sun J, Zhao G.
A quadratically convergent polynomial long-step algorithm for A class of nonlinear monotone complementarity problems. (2000).