Bima Journal of Science and Technology, Vol. 9(1A) Mar, 2025 ISSN: 2536-6041

DOI: 10.56892/bima.v9i1A.1258

Accelerated ADMM Algorithm For Solving Games Theory with Mixed Strategies

Kazeem Dawodu*

Department of Mathematical Sciences, Federal University of Technology, Akure,

PMB 704, Ondo-State, Nigeria.

Corresponding Author: kadawodu@futa.edu.ng

ABSTRACT

This study presents the Alternating Direction Method of multipliers (ADMM) for solving linear programming problem (LPP) which is also known as *proximal point algorithm.* The ADMM was deployed because of its strong convergence properties of the method of multipliers, the decomposability property of dual ascent and the potential to solve large- scale structured optimization problems. The update formulas for the LPP were derived from the associated augmented Lagrangian with the primal and dual residuals also derived for the convergence of the algorithm. The Game theory was re-structured into a LPP amenable to the ADMM with a derived matrix operator that is invertible to guarantee its convergence. Numerical examples were simulated to ascertain the performance of the method in terms of speed and accuracy.

AMS subject classifications: 49, 65, 90

Keywords: ADMM, primal and dual convergences, Accelerator-variant,

INTRODUCTION

Another line of research has focused on enhancing the performance and convergence of ADMM for linear optimization problems. For example, Qin et al. (2017) proposed a new primal-dual method for large-scale linear programming with linear inequality constraints. The study demonstrated that the ADMM can achieve faster convergence and better scalability compared to the penalty methods particularly for large-scale linear optimization problems. Furthermore, several studies have investigated the application of ADMM to specific classes of linear optimization problems. For instance, Li et al. (2019) proposed an ADMM algorithm for solving linear programs with box constraints. The authors developed a specialized ADMM scheme that exploits the structure of box constraints to achieve faster convergence and better scalability. The study demonstrated the effectiveness of the proposed algorithm on various practical applications, such as image reconstruction and compressed sensing.

Famaey et. al, (2014) focused on the application of ADMM to the LASSO (Least Absolute Shrinkage and Selection Operator) problem. It introduces an accelerated version of ADMM that improves the convergence speed, and provides convergence analysis and experimental results for both centralized and distributed LASSO problems. Yuan and Wang (2017) provided an accelerated ADMM for linear and convex quadratic network utility maximization and then proposed an accelerated version of ADMM for solving network utility maximization problems in signal processing. It explores the convergence properties and computational efficiency of the accelerated ADMM, and provides numerical experiments to demonstrate its performance. Patrascu and Rosu (2019) proposed an ADMM-based approach for solving linear programing problems with both equality and inequality constraints. It discusses the convergence properties and computational efficiency of the algorithm,

DOI: 10.56892/bima.v9i1A.1258

and provides numerical experiments to demonstrate its effectiveness. Guo et.al., (2019) derived an adaptive Nesterov-type accelerated version of ADMM for solving linearly constrained convex optimization problems. It discusses the theoretical properties and numerical performance of the algorithm, and provides comparative experiments with other algorithms. Hong and Sun (2016) discussed the implementation of primal-dual methods, including ADMM, for solving large-scale linear programming problems. It provides a detailed description of the algorithmic framework for ADMM and discussed the practical issues that arose in implementing the method. Yang and Zhang (2017) used the alternating direction method for solving semidefinite programming problems with linear constraints. It shows that the proposed algorithm is equivalent to a variant of ADMM and provides convergence analysis and numerical experiments.

In the work of Dawodu (2021) on Optimal control problems with multiple delay, the ADMM was accelerated with a parameter factor in the sense of Nesterov (1983). The method presented in the study was used to find a numerical solution to the Optimal Control model constrained by Partial Differential Equation. A one-dimensional heat equation optimization problem driven by a partial differential equation was solved using the ADMM tool. The rate of convergence of the model for increasing iterations was determined by deriving the primal-dual residuals for the effectiveness and level of accuracy of the problem using the algorithm.

MATERIALS AND METHODS

Statement of Problem

Considering the generalized linear programming problem (LPP)

$$\min_{x,u}\left(\sum_{i=1}^{N} c_i x_i + \sum_{j=1}^{M} d_j u_j\right) \quad (1)$$

$$\text{s.t.}\ \sum_{i=1}^{N} a_{ij} x_i + \sum_{j=1}^{M} b_{ij} u_j \leq e_i, \quad i = 1,2,\ldots,N; \quad j = 1,2,\ldots,M \quad (2)$$

where $x \in \mathbf{R}^N$, $u \in \mathbf{R}^M$, $c_i, d_j, a_{ij}, b_{ij} \in \mathbf{R}^+$.

Expanding Eqns. (1) and (2) yields the compact equation written below as

$$\min(C^T x + D^T u)\ \ s.t\ \ Ax + Bu \leq E, \quad (3)$$

where,
$x = (x_1, x_2, \cdots x_N)^T \in \mathbf{R}^N$,
$u = (u_1, u_2 \cdots u_M)^T \in \mathbf{R}^M$,
$C = (c_1, c_2, \cdots, c_N)^T \in \mathbf{R}^N$,
$D = (d_1, d_2, \cdots, d_M)^T \in \mathbf{R}^M$,
$E = (e_1, e_2, \cdots, e_N)^T \in \mathbf{R}^N$,
$A \in \mathbf{R}^{N\times N}$ and $B \in \mathbf{R}^{N\times M}$ with entries of A and B describe as $a_{ij} \in A$ and $b_{ij} \in B$ respectively for $i = 1,2\ldots N$; $j = 1,\ldots M$. The model above is a linear programming problem amenable to the ADDM since the objective function is separable in x and u, closed and convex; while the constraint is linear and coupled in both variables in x and u. The compact form of eqns. (1) and (2) expressed in eqns. (3) was subjected to the ADMM before imposing the karush-Khurn-Tucker(KKT) optimality condition. It was later accelerated using the Gauss-Seidel accelerator-variant to speed up the rate of convergence of the ADMM algorithm.

DOI: 10.56892/bima.v9i1A.1258

Implementation of ADMM on LPP

Given the optimization problem below

$$\min\ (C^T x + D^T u)\quad s.t\quad Ax + Bu \le E, \tag{4}$$

then the associated augmented Lagrangian of eqn. (4) is

$$\min_{x,u} L_\rho(x,u,s,\lambda) = C^T x + D^T u + \lambda^T(Ax + Bu - E + s) + \frac{\rho}{2} \parallel Ax + Bu - E + s\parallel_2^2 + l_+(z), \tag{5}$$

where $s \ge 0$, λ is the Lagrange multiplier, $\rho \ge 0$ is the penalty parameters, $\parallel . \parallel_2$ is the euclidean (spectral) norms of a vector (matrix) arguments, s is the introduced slack vector and $l_+$is the indicator function for the non-negative orthants defined as

$$l_+(z) = \begin{cases} 0 & for\ z \ge 0 \\ \infty & otherwise \end{cases}$$

The Lagrangian formulation is derived thus:

$$C^T x + D^T u + \lambda^T(Ax + Bu - E + s + v) + \frac{\rho}{2} \parallel Ax + Bu - E + s + v\parallel_2^2 \tag{6}$$

compactly written as

$$F + \lambda^T w + \frac{\rho}{2} \parallel w\parallel_2^2 = F + \frac{\rho}{2} \parallel w + v\parallel_2^2 - \frac{\rho}{2} v^T v \qquad s \ge 0,$$

where $F = C^T x + D^T u$, $w = Ax + Bu - E + s$ and upon expansion and collections of like-terms yields

$$F + \lambda^T w + \frac{\rho}{2} w^T w = F + \frac{\rho}{2} w^T w + \rho v^T w, \tag{7}$$

$$\lambda^T w = \rho v^T w \tag{8}$$

$$v = \frac{\lambda^T}{\rho}, \tag{9}$$

where $\frac{\rho}{2} v^T v$ is a constant. The scale augmented Lagrangian function is now

$$\min L_\rho(x,u,s,v) = C^T x + D^T u + \frac{\rho}{2} \parallel Ax + Bu - E + s + v\parallel_2^2 ;\ s \ge 0, \tag{10}$$

where $v = \frac{\lambda}{\rho}$ is the scaled dual variable.

The following Karush-Khun-Tucker(KKT) optimality conditions will then be imposed on the scaled augmented lagrangian for the derivation of the ADMM update formulas.

$$0 = \nabla_x L_\rho(x, u^k, s^k, v^k)$$
$$= \nabla_u L_\rho(x^{k+1}, u, s^k, v^k) = \nabla_s L_\rho$$
$$= (x^{k+1}, u^{k+1}, s, v^k), \tag{11}$$

and

$$v^{k+1} = v^k + \left(Ax^{k+1} + Bu^{k+1} - E + s^{k+1}\right) \tag{12}$$

for $v = \frac{\lambda}{\rho}$. Applying the optimality conditions above to the separable and convex operators of the objective function and linear constraints given by the sequential minimization of x, u, s and v in the Lagrangian function in eqn. (12) yields the update formulas below.

Update on-x:

$$0 = \nabla_x L_\rho(x, u^k, s^k, v^k)$$

Bima Journal of Science and Technology, Vol. 9(1A) Mar, 2025 ISSN: 2536-6041
DOI: 10.56892/bima.v9i1A.1258

$= \frac{\partial L_\rho}{\partial x}[C^T x + D^T u^k + \frac{\rho}{2} \parallel Ax + Bu^k - E + s^k + v^k\|_2^2]$ (1
$= C + \rho A^T(Ax + Bu^k - E + s^k + v^k).$ (1
This implies that
$x^{k+1} =-(A^TA)^{-1}[\rho^{-1}C + A^T(Bu^k - E + s^k + v^k)]$ (1

Update on-u:

Considering the update of the u vector,we then set
$0 = \nabla_u L_\rho(x^{k+1}, u, s^k, v^k)$
$= \frac{\partial L_\rho}{\partial u}[C^T x^{k+1} + D^T u + \frac{\rho}{2} \parallel Ax^{k+1} + Bu - E + s^k + v^k\|_2^2]$ (1
$= D + \rho B^T(Ax^{k+1} + Bu - E + s^k + v^k).$
This implies that
$u^{k+1} =-(B^TB)^{-1}[\rho^{-1}D + B^T(Ax^{k+1} - E + s^k + v^k).$ (1
Introducing the acceleration or relaxation factor (parameter) $\alpha \in [1.8,\ 2]$ in the sense of Nesterov (1983) in the u-update formula by replacing Ax^{k+1} by $h^{k+1} = Ax^{k+1} + (1-\alpha)(Bu^k - E + s^k)$ in eqn. (18) yields
$u^{k+1} =-(B^TB)^{-1}[\rho^{-1}D + B^T(h^{k+1} - E + s^k + v^k).$ (1

Update on-s:

Considering the update of the slack variable s leads to
$0 = \nabla_s L_\rho(x^{k+1}, u^{k+1}, s, v^k)$
$= \frac{\partial L_\rho}{\partial s}[C^T x^{k+1} + D^T u^{k+1} + \frac{\rho}{2} \parallel Ax^{k+1} + Bu^{k+1} - E + s + v^k\|_2^2],$ (20)
$0 = (Ax^{k+1} + Bu^{k+1} - E + s + v^k).$
This implies that
$s^{k+1} =-(Ax^{k+1} + Bu^{k+1} - E + v^k); \quad s \geq 0$
$s^{k+1} = \max\{0,\ -(h^{k+1} + Bu^{k+1} - E + v^k)\};$ Component wise (22)

Update on-v:

Updating the dual variable v requires that
$v^{k+1} = v^k + (Ax^{k+1} + Bu^{k+1} - E + s^{k+1}).$ (23)
Replacing Ax^{k+1} with h^{k+1} in eqn. (23) yields
$v^{k+1} = v^k + (h^{k+1} + Bu^{k+1} - E + s^{k+1}).$ (24)
Summarily, the over-relaxed ADMM for the original problem is given as;

$$
\begin{aligned}
x^{k+1} &=-(A^TA)^{-1}[\rho^{-1}C + A^T(Bu^k - E + s^k + v^k),\\
u^{k+1} &=-(B^TB)^{-1}[\rho^{-1}D + B^T(h^{k+1} - E + s^k + v^k),\\
s^{k+1} &= \max\{0,\ -(h^{k+1} + Bu^{k+1} - E + v^k)\},\\
v^{k+1} &= v^k + (h^{k+1} + Bu^{k+1} - E + s^{k+1}). \quad (25)
\end{aligned}
$$

Given the starting points (x^0, u^0) and initial starting points for the adjoint and slack variables ($\lambda^0,\ s^0$), the ADMM solves the problem (4) by the update schemes in eqn. (25) above.

Convergence analysis and Profile of the ADMM on LPP

In the development of the ADMM for the multiobjective linear programming problems, the convergence of the ADMM to an optimal solution is guaranteed provided the solution exists and the matrix operators are well-posed (invertible, consistent and stable). However, the limit of the proposed ADMM iterates usually satisfy the set of first-order optimality condition by producing a certificate of either *primal or dual feasibility* or both. In the derivation of the convergence residues, we assumed that the multi-objective function of the problem is represented as $p^k = f(x^k) + g(u^k)$ as k^{th} iteration (cycle) approaches the optimal objective value $p^*(\,i.e\ \ p^k \to p^*)$ for large value of $k\ (i.e\ k \to \infty)$ known as the optimal objective value where the dual residual generated by each iteration converges to zero. In same light, we assumed $r^k = Ax^k + Bu^k - E$, being the primal residual of the constraint at each iteration,

Bima Journal of Science and Technology, Vol. 9(1A) Mar, 2025 ISSN: 2536-6041

DOI: 10.56892/bima.v9i1A.1258

approaches zero as the algorithm approaches optimality.

Theorem 1

Suppose p^k is the objective function value of the convex and sub-differentiable functions f and g such that it converges to the optimal objective value p^* for the given constraints $Ax + Bu - E \leq 0$ and multiplier λ, then there exists dual residual $||d^{k+1}|| = \rho A^T\left[B\left(u^{k+1} - u^k\right) + \left(s^{k+1} - s^k\right)\right]$ that converges to zero for a given penalty parameter ρ.

Proof:

Given the objective function $p^k = f(x^k) + g(u^k)$ and the linear inequality constraint $Ax + Bu \leq E$, the associated Lagrangian with **slacks** is stated thus

$$L_\rho(x, u, s, \lambda) = f(x) + g(u) + \lambda^T(Ax + Bu - E + s) + \frac{\rho}{2} \| Ax + Bu - E + s\|_2^2.$$

Applying the optimality conditions (KKT), we have

$$\partial f(x) + A^T\lambda^k + \rho(A^TAx + A^TBu - A^TE + A^Ts) = 0$$

$$\partial f(x^{k+1}) + A^T\lambda^k + \rho A^T(Ax^{k+1} + Bu^k - E + s^k) = 0,$$

$$\partial f(x^{k+1}) + A^T\lambda^k + \rho A^T(\underbrace{Ax^{k+1} + Bu^{k+1} - E + s^{k+1}}_{r^{k+1}}),$$

$$-\rho A^TBu^{k+1} - \rho A^Ts^{k+1} + \rho A^TBu^k + \rho A^Ts^k = 0,$$

where the **primal residual** is expressed as

$$r^{k+1} = (Ax^{k+1} + Bu^{k+1} - E + s^{k+1}). \tag{26}$$

Therefore,

$$\partial f\left(x^{k+1}\right) + A^T\lambda^k + \rho A^Tr^{k+1} - \rho A^TB\left(u^{k+1} - u^k\right) - \rho A^T\left(s^{k+1} - s^k\right) = 0, \tag{27}$$

$$\rho A^T\left[B\left(u^{k+1} - u^k\right) + \left(s^{k+1} - s^k\right)\right] = \partial f\left(x^{k+1}\right) + A^T\left[\underbrace{\lambda^k + \rho r^{k+1}}_{\lambda^{k+1}}\right], \tag{28}$$

since at the ADMM, the update $\partial f(x^{k+1}) + A^T\lambda^{k+1} \to 0$, then its **dual residual**

$$||d^{k+1}|| = \rho A^TB[(u^{k+1} - u^k) + (s^{k+1} - s^k)] \to 0. \tag{29}$$

which completes the proof. Q.E.D

The convergence of the dual and primal feasibility to zero in the equation above is a clear indication that the algorithm has **super linear convergence.** However, the LPP in eqn. (3) can be re-structured thus:

$$\min\ Q^Tz\ \ s.t\ \ Pz \leq E \tag{30}$$

where $z = [x \quad y]^T \in \mathbf{R}^{N+M}$, $Q = [C^T \quad D^T] \in \mathbf{R}^{(N+M)\times 1}$, $P = [A \quad B] \in \mathbf{R}^{(N)\times(N+M)}$ and $E \in \mathbf{R}^{N\times 1}$.

Bima Journal of Science and Technology, Vol. 9(1A) Mar, 2025 ISSN: 2536-6041

DOI: 10.56892/bima.v9i1A.1258

Theorem 2

Given a linear programming problem (LPP)

$$\min \frac{1}{2} Q^T z \ such\ that\ Pz \leq E \quad (31)$$

where $Q \in \mathbf{R}^{(N+M)\times 1}, z \in \mathbf{R}^{(N+M)\times 1}$. $P \in \mathbf{R}^{N\times(N+M)}$ and $E \in \mathbf{R}^{N\times 1}$ then the optimal stepsize for the LPP is

$$\rho^* = [\sqrt{\lambda_{\min}(PQ^{-1}P^T)\lambda_{\max}(PQ^{-1}P^T)}]^{-1} \quad (3$$

and the convergence factor.

$$\xi^* = \frac{\lambda_{\max}(PQ^{-1}P^T)-\sqrt{\lambda_{\min}(PQ^{-1}P^T)\lambda_{\max}(PQ^{-1}A^T)}}{\lambda_{\max}(PQ^{-1}A^T)+\sqrt{\lambda_{\min}(PQ^{-1}P^T)\lambda_{\max}(PQ^{-1}P^T)}} \quad (33)$$

for $\alpha \in (0,2)$, $\lambda_{\min}(PQ^{-1}P^T)$ and $\lambda_{\max}(PQ^{-1}P^T)$ are minimum and maximum eigenvalue of $PQ^{-1}P^T$ respectively.

See proof in Ghadimi et. al. (2014).

The result of the optimal parameter selection of the LPP above is also stated and proven in Ghadimi (2014) where $P \in \mathbf{R}^{N\times N}$ and $Q \in \mathbf{R}^{(N+1)\times(N+1)}$

Stopping criteria

The reasonable termination (stopping) criteria for the ADMM are to select the primal and dual residuals, so small, such that $\| r^{k+1}\|_2 \leq \epsilon^{prim}$ and $\| d^{k+1}\|_2 \leq \epsilon^{dual}$, where $\epsilon^{prim} > 0$ and $\epsilon^{dual} > 0$ are the tolerances of the primal and dual feasibility conditions respectively for the convergences of the ADMM. However,the choices of our tolerances depend on both the relative and absolute criteria on account that the ℓ_2 norms are in $\mathbf{R}^M$ or $\mathbf{R}^N$. In Boyd *et. al* (2011), the computation of the relative tolerance, absolute tolerance, primal and dual residuals for the ADMM on the primal were adopted and given as $\epsilon^{rel} = 10^{-3}, \epsilon^{abs} = 10^{-4}, \epsilon^{prim} = \sqrt{M}\epsilon^{abs} + \epsilon^{rel}\max\{ \| Ax^{k+1}\|_2, \| s^{k+1}\|_2\}$ and $\epsilon^{dual} = \sqrt{M}\epsilon^{abs}$ respectively. The dual is then computed as: $\epsilon^{prim} = \sqrt{N}\epsilon^{abs} + \epsilon^{rel}\max\{ \| By^{k+1}\|_2, \| s^{k+1}\|_2\}$ and $\epsilon^{dual} = \sqrt{N}\epsilon^{abs}$, $B \in \mathbf{R}^{(N-1)\times(N+1)}$ and $C \in \mathbf{R}^{(N-1)\times 1}$.

ADMM Algorithm for LPP

Step 1 :***(Initialization)***

Input Initial values $x_i^0,\ u_i^0, \lambda_i^0,\ \epsilon^{prim} > 0\ , \epsilon^{dual} > 0\ \ M,\ T,\ \rho$

Step 2: ***(Update formulas)*** $k = 0,1,\cdots,N$

$x^{k+1} = -(A^TA)^{-1}[\rho^{-1}C + A^T(Bu^k - E + s^k + v^k)]$ eqn. (15)

$u^{k+1} = -(B^TB)^{-1}[\rho^{-1}D + B^T(h^{k+1} - E + s^k + v^k)$ eqn. (19)

$v^{k+1} = v^k + \left(Ax^{k+1} + Bu^{k+1} - E + s^{k+1}\right)$ eqn. (23)

Step 3: ***(Compute primal and dual residuals)*** $k = 1,2\cdots,N$

$r^{k+1} = (Ax^{k+1} + Bu^{k+1} - E + s^{k+1})$ eqn. (26)

$d^{k+1} = \rho A^TB\left[\left(u^{k+1} - u^k\right) + \left(s^{k+1} - s^k\right)\right]$ eqn. (29)

Step 4: ***(Compute primal and dual Tolerance values)*** $k = 1,2\cdots,N$

$\epsilon^{prim} = \sqrt{M}\epsilon^{abs} + \epsilon^{rel}\max\{ \| Ax^{k+1}\|_2, \| Bu^{k+1}\|_2, \| s^{k+1}\|_2\}$

$\epsilon^{dual} = \sqrt{M}\epsilon^{abs}$

Step 5:***(Termination criteria)***

Stop if $||r^{k+1}|| \leq \epsilon^{prim}$ *and* $\left|\left|d^{k+1}\right|\right| \leq \epsilon^{dual}$

Output $x^* = x^{k+1},\ u^* = u^{k+1}$, and

End function otherwise repeat step 2 for $k = k + 1$

Bima Journal of Science and Technology, Vol. 9(1A) Mar, 2025 ISSN: 2536-6041

DOI: 10.56892/bima.v9i1A.1258

ADMM Implementation on Game theory

Game theory bears a strong relationship with linear programming since any two-person zero sum game can be expressed as a linear programme. In Dantzig (1963), it was stated that J. Von-Neumann in 1947 introduced the Simplex method in solving games by expressing it with the concept of duality in linear programming.

A \ B	y_1	y_1	…	y_N
x_1	a_{11}	a_{11}	…	a_{11}
x_2	a_{21}	a_{22}	…	a_{2N}
…				
x_M	a_{M1}	a_{M2}	…	a_{MN}

In the *payoff matrix* (table) above, player A has M strategies with $x_1, x_2 \cdots x_M$ as the optimal probability and player B has N strategies with $y_1, y_2, y_3 \cdots y_N$ as the optimal strategies where $0 \leq x_i\ y_j \leq 1$, $\forall i, j$.

LPP formulation for the Primal

The linear programming models for determining the optimal strategies for the primal can be solved by the maximum LPP model :-

$$\max_{x_i} [\min \left\{ \sum_{i=1}^{M} a_{i1} x_i, \sum_{i=1}^{M} a_{i2} x_i, \cdots, \sum_{i=1}^{M} a_{iN} x_i \right\}$$

$$x_1 + x_2 + x_3 + \cdots + x_M = 1 \qquad (34)$$

$$x_i \geq 0,\ i = 1,2, \cdots M$$

Let

$$\mathrm{v} = \min \left\{ \sum_{i=1}^{M} a_{i1} x_i, \sum_{i=1}^{M} a_{i2} x_i, \cdots, \sum_{i=1}^{M} a_{iN} x_i \right\}$$

then it implies that any element in the bracket satisfies the equation

$$\mathrm{v} - \sum_{i=1}^{M} a_{ij} x_i \leq 0, \quad j = 0,1,..,N. \qquad (35)$$

Therefore, the linear programming problem of the primal can be expressed as

$$\max z = v \qquad (36)$$

$$\text{s.t} \, \mathrm{v} - \sum_{i=1}^{M} a_{ij} x_i \leq 0, \quad j = 0,1,.,N. \qquad (37)$$

$$\sum_{i=1}^{M} x_i = 1 \qquad (38)$$

$$x_i \geq 0, \quad \forall i = 0,1,..,M. \qquad (39)$$

$$v \quad unrestricted, \qquad (40)$$

where v is the value of game and it is unrestricted in signs. Expanding the objective function in eqn. (36) yields $C^T x$ and upon expansion of constraint eqns. (37) to (40) for various values of i and j yields the constructed matrix operator below:

$$\begin{bmatrix} 1 & | & -a_{11} & a_{21} & \cdots & -a_{M1} \\ 1 & | & -a_{12} & a_{22} & \cdots & -a_{M2} \\ \vdots & | & \vdots & \vdots & \vdots & \vdots \\ 1 & | & -a_{1N} & a_{2N} & \cdots & -a_{MN} \\ 0 & | & 1 & 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} v \\ x_1 \\ x_2 \\ \vdots \\ x_M \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 1 \end{bmatrix} \qquad (41)$$

$$\begin{bmatrix} & | & \\ & | & \\ \bar{h} & | & -P^T \\ & | & \\ 0 & | & \bar{q}^T \end{bmatrix} \begin{bmatrix} v \\ x_1 \\ x_2 \\ \vdots \\ x_M \end{bmatrix} + \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_N \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 1 \end{bmatrix}, \qquad (42)$$

Compactly written as

$$Ax + s = \alpha, \qquad (43)$$

DOI: 10.56892/bima.v9i1A.1258

where $A \in \mathbf{R}^{(N+1)\times(M+1)}$ is invertible, $x = (v, x_1, x_2, \cdots x_M) \in \mathbf{R}^{(M+1)}$, $s = (s_1, s_2, \cdots s_N, 0) \in \mathbf{R}^{(N+1)}$, $\alpha = (0,0,\cdots,1) \in \mathbf{R}^{(N+1)}$, $\bar{h} = (1,1,\cdots,1) \in \mathbf{R}^N$, $\bar{q} = (1,1,\cdots,1) \in \mathbf{R}^M$, $P \in \mathbf{R}^{M\times N}$ $C = (1\ \bar{\bar{0}}) \in \mathbf{R}^{(M+1)}$, and P is the payoff matrix. The game can now be presented in the form amenable to ADMM as expressed below:

$$\min z = C^T x \qquad (44)$$

$$s.t\ \ Ax + s = \propto\ \ \geq 0,\ x_i \geq 0, \forall x_i \in \boldsymbol{x}\ , \forall s_i \in \boldsymbol{s};\ i = 1,2,\cdots,M \qquad (45)$$

$v\ \ unrestricted$

LPP formulation for the Dual

Similarly, the linear programming model for the dual with optimal strategies can be solved by the minimum problem by the duality principle.

$$\min w = v \qquad (46)$$

$$s.t\ \ v - \sum_{j=1}^{N} a_{ij} y_j \geq 0,$$

$i = 1,2,3,\cdots,M$ (4

$\sum_{i=1}^{N} y_j = 1$ (4

$y_j \geq 0,\ \forall j = 1,2,3\cdots,N;$ (4

$v\ \ \ unrestricted$ (50)

where v (the game value) is unrestricted in signs and expressed as

$v = \max\{\sum_{j=1}^{N} a_{1j} y_{j,} \cdots, \sum_{j=1}^{N} a_{Mj} y_j\}$

(51)

,wherefore any element in the bracket is such that $v - \sum_{j=1}^{N} a_{ij} y_j, \geq 0,\ \ for\ \ i = 1,2,3\cdots,M.$

Similarly, expanding the objective function in eqn. (46) yields $D^T y$ and upon expansion of constraint eqns. (47) to (50) for various values of i and j yields the matrix formulation below for the player B optimal strategies.

$$\begin{bmatrix} 1 & | & -a_{11} & a_{12} & \cdots & -a_{1N} \\ 1 & | & -a_{21} & a_{22} & \cdots & -a_{2N} \\ \vdots & | & \vdots & \vdots & \vdots & \vdots \\ 1 & | & -a_{M1} & a_{M2} & \cdots & -a_{MN} \\ 0 & | & 1 & 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} w \\ y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \geq \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 1 \end{bmatrix} \qquad (52)$$

$$\begin{bmatrix} & | & \\ & | & \\ \bar{q} & | & -P \\ & | & \\ 0 & | & \bar{h}^T \end{bmatrix} \begin{bmatrix} w \\ y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} - \begin{bmatrix} \bar{s}_1 \\ \bar{s}_2 \\ \vdots \\ \bar{s}_M \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 1 \end{bmatrix}, \qquad (53)$$

compactly written as

$$By - s = \bar{\alpha}, \qquad (54)$$

where $B \in \mathbf{R}^{(M+1)\times(N+1)}$ is invertible, $y = = (w, y_1, y_2, \cdots y_N) \in \mathbf{R}^{(N+1)}$, $s = (\bar{s}_1, \bar{s}_2, \cdots, \bar{s}_M, 0) \in \mathbf{R}^{(M+1)}$, $D = (1\ \hat{0}) \in \mathbf{R}^{(N+1)}$, $\hat{0} = (0,0,\cdots,0) \in \mathbf{R}^N$ and $\bar{\alpha} = (0,0,\cdots,1) \in \mathbf{R}^{M+1}$. The game can now be presented in the form amenable to ADMM as expressed below:

$$\max_y w = D^T y$$

$$s.t\ \ By - s = \bar{\alpha}$$

$$y_j \geq 0,\ s_j \geq 0,\ \ \forall y_j \in y, \forall s_j \in s; j = 1,2,3 \qquad (55)$$

$v\ \ \ unrestricted$

DOI: 10.56892/bima.v9i1A.1258

ADMM formulation

For the Primal linear program, the Lagrangian is formulated thus

$$L_\rho(x,s,\gamma) = C^T x + \frac{\rho}{2} \| Ax + s - \alpha + \gamma \|_2^2,\ x \geq 0,\ s \geq 0,\ \gamma = \frac{\lambda}{\rho}. \quad (56)$$

Applying the necessary conditions in eqn. (11) to (56) and replacing Ax^{k+1} with $h^{k+1} = \alpha Ax^{k+1} + (1-\alpha)s^{k+1}$ yields the following accelerated update formulas for

$$x^{k+1} = \max\{0,\ -(A^TA)^{-1}(\rho^{-1}C - A^T(s^k + \gamma^k - \alpha)$$

$$s^{k+1} = \max\{0,\ -(h^{k+1} + \gamma^k - \alpha)\}$$

$$\gamma^{k+1} = \gamma^k + (h^{k+1} + s^{k+1} - \alpha) \quad (57)$$

$$\gamma^{k+1} = \gamma^k + (h^{k+1} + s^{k+1} - \alpha)\ .$$

For dual linear program, the Lagrangian is formulated thus:

$$L_\rho(y,s,\phi) = D^T y + \frac{\rho}{2} \| By - s - \overline{\alpha} + \phi \|_2^2,\ y \geq 0,\ s \geq 0,\ \phi = \frac{\lambda}{\rho}. \quad (58)$$

and the corresponding accelerated update formulas are given as;

$$y^{k+1} = \max\{0,\ -(B^TB)^{-1}(\rho^{-1}D - B^T(s^k + \overline{\alpha} - \phi^k)\}$$

$$s^{k+1} = \max\{0,\ (h^{k+1} - \overline{\alpha} + \phi^k)\}$$

$$\phi^{k+1} = \phi^k + (h^{k+1} - s^{k+1} - \overline{\alpha}) \quad (59)$$

$$y \geq 0,\quad s \geq 0,\quad \forall y_j \in y, \forall \overline{s}_j \in s;\ j = 1,2,\cdots,$$

where By^{k+1} is being replaced by $h^{k+1} = \alpha By^{k+1} + (1-\alpha)s^{k+1}$.

Primal Convergence Analyses

For the Primal program, let $f(x^{k+1}) = p^{k+1}$ such that $p^{k+1} \to p^*$ at optimum point x^{k+1}. Then

$$\partial f(x^{k+1}) + A^T\lambda^k + \rho A^T(Ax^{k+1} + s^k - \alpha) = 0$$

$$\partial f(x^{k+1}) + A^T\lambda^k + \rho A^T\underbrace{(Ax^{k+1} + s^{k+1} - \alpha)}_{r^{k+1}}$$

$$+\rho A^T s^k - \rho A^T s^{k+1} = 0$$

$$\partial f(x^{k+1}) + A^T[\underbrace{\lambda^k + \rho r^{k+1}}_{\lambda^{k+1}}] - \rho A^T(s^{k+1} - s^k) = 0$$

$$\rho A^T(s^{k+1} - s^k) = \partial f(x^{k+1}) + A^T\lambda^{k+1},$$

where $r^{k+1} = Ax^{k+1} + s^{k+1} - \alpha$ is the primal residual. If $r^{k+1} = Ax^{k+1} + s^{k+1} - \alpha \leq \epsilon^{prim}$ then at convergence, the dual residual (iterates) at each cycle becomes

$$d^{k+1} = \rho A^T(s^{k+1} - s^k) = \partial f(x^{k+1}) + A^T\lambda^{k+1} \leq \epsilon^{dual} \quad (60)$$

Similarly, for the dual program, the primal residual is $\overline{r}^{k+1} = By^{k+1} - s^{k+1} - \overline{\alpha}$ and for $\partial f(y^{k+1}) + B^T\phi^{k+1} \to 0$ the dual residual at optimum point y^{k+1} then becomes

$$\overline{d}^{k+1} = \rho B^T(s^{k+1} - s^k) = \partial f(y^{k+1}) + A^T\lambda^{k+1} \leq \epsilon^{dual} \quad (61)$$

For further analysis, from Boyd et.al (2011),

$$\| r^{k+1} \| = \epsilon^{prim} \leq \epsilon^{dual} + \| Ax^{k+1} + s^{k+1} \|$$

$$\leq \epsilon^{dual} + \epsilon^{rel}\max\{ \| Ax^{k+1} \|,\ \| s^{k+1} \| \}$$

$$\leq \epsilon^{dual} + \epsilon^{rel}\max\{ \| Ax^{k+1} \|, \| s^{k+1} \| \} \quad (62)$$

$$where \quad \| d^{k+1} \| = \epsilon^{dual} \leq \sqrt{M}\epsilon^{abs} \quad (63)$$

Bima Journal of Science and Technology, Vol. 9(1A) Mar, 2025 ISSN: 2536-6041
DOI: 10.56892/bima.v9i1A.1258

ADMM Algorithm on the Primal LPP for Game Theory

ADMM Algorithm for LPP
Step 1 :***(Initialization)*** Input Initial values $x_i^0,\ u_i^0, \lambda_i^0,\ \epsilon^{prim} > 0\ \ , \epsilon^{dual} > 0\ \ M,\ T,\ \rho$ Step 2: ***(Update formulas)*** $k = 0,1,\cdots,N$ $x^{k+1} = \max\{0, -(A^TA)^{-1}(\rho^{-1}A^T + s^k + \gamma^k)\}$ eqn. (15) $h^{k+1} = \alpha Ax^{k+1} + (1-\alpha)s^k$ eqn. (19) $s^{k+1} = \max\{0, -(\rho^{-1}I + h^{k+1} + \gamma^k)\}$ eqn. (23) $\gamma^{k+1} = \gamma^k + (h^{k+1} + s^{k+1})$ Step 3: ***(Compute primal and dual residuals)*** $k = 1,2\cdots,N$ $r^{k+1} = Ax^{k+1} + s^{k+1}$ eqn. (26) $d^{k+1} = \rho A^T(s^{k+1} - s^k)$ eqn. (29) Step 4: ***(Compute primal and dual Tolerance values)*** $k = 1,2\cdots,N$ $\epsilon^{prim} = \sqrt{M}\epsilon^{abs} + \epsilon^{rel}\max\{\ \Vert Ax^{k+1}\Vert_2, \Vert Bu^{k+1}\Vert_2, \Vert s^{k+1}\Vert_2\}$ $\epsilon^{dual} = \sqrt{M}\epsilon^{abs}$ Step 5:***(Termination criteria)*** ***Stop if*** $\|r^{k+1}\| \le \epsilon^{prim}$ $\quad and \quad$ $\|d^{k+1}\| \le \epsilon^{dual}$ ***Output*** $x^* = x^{k+1},\ u^* = u^{k+1}$, and ***End function*** otherwise repeat step 2 for $k = k+1$

Numerical Examples
Example 1:

$\min 3x + 2u$
$s.t 2 - u \le 6$
$x + 2u \le 1$
$x\ \ u \ge 0.$

The above problem can be re-structured into matrix form below:

$\min C^Tx \ \ s.t\ \ Ax = b;\ \ x \ge 0,$ (64)

where $x = (x\ \ u),\ C = (3\ \ 2)^T$,
$b = (6\ \ 1)^T,\ \ and\ \ A = \begin{bmatrix}(2 & -1 \\ 1 & 2\end{bmatrix}$

DOI: 10.56892/bima.v9i1A.1258

iter. k	x-norm $\|x\|$	u-norm $\|u\|$	Primal residue $\|r^{k+1}\|$	Dual residue $\|d^{k+1}\|$	Primal conv. $\|r^{k+1} - r^k\|$	Dual conv. $\|d^{k+1} - d^k\|$
1	2.2137	2.0105	0	0	4.9499	0
2	3.0312	2.9612	1.348	5.9861	2.2153	0
3	2.9157	2.8382	0.8767	4.1845	0.2790	0
4	2.6833	2.5992	0.6088	4.3173	0.5344	0
5	2.6353	2.5496	0.2984	4.3075	0.1108	0
6	2.6689	2.5843	0.3187	4.3082	0.0775	0
7	2.6896	2.6056	0.3002	4.3082	0.0478	0
8	2.6890	2.6051	0.2984	4.3082	0.0012	0
9	2.6845	2.6004	0.2984	4.3082	0.0105	0
10	2.6832	2.5991	0.2981	4.3082	0	0

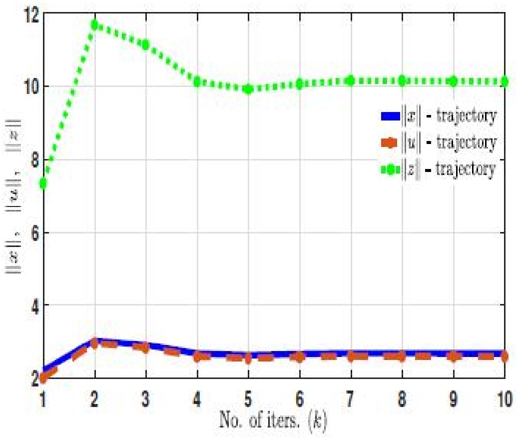


Figure 1: Trajectories of the Primal (Dual) variables (||x||, ||u||) and the Obj. function (||z||).

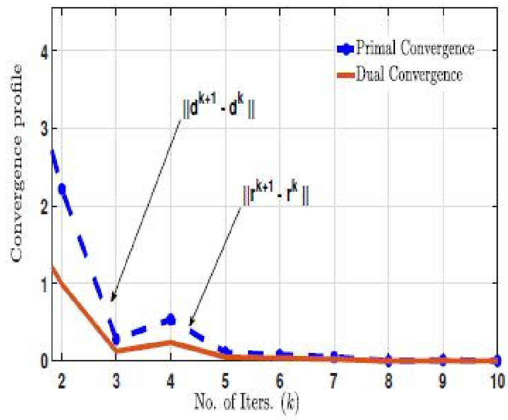


Figure 3: Primal and Dual Convergences of the ADMM Algorithm as $||r^{k+1} - r^k|| \to 0$ and $||d^{k+1} - d^k|| \to 0$

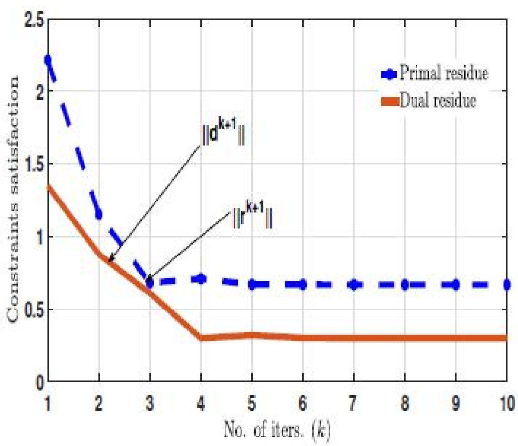


Figure 2: Primal and Dual residues towards convergence.

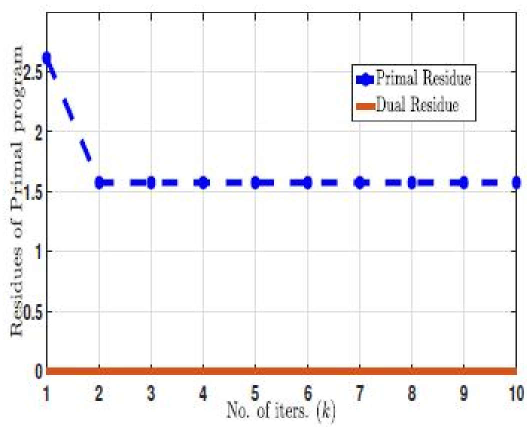


Figure 4: Primal $||r^{k+1}||$ and Dual $||d^{k+1}||$ residues.

Bima Journal of Science and Technology, Vol. 9(1A) Mar, 2025 ISSN: 2536-6041

DOI: 10.56892/bima.v9i1A.1258

Example 2:

Solve the following game by ADMM. The value of the game, v, lies between -2 and 2.

$$P = \begin{bmatrix} 3 & -1 & -3 \\ -2 & 4 & -1 \\ -5 & -6 & 2 \end{bmatrix} \qquad (65)$$

Player A mixes strategies $A1$, $A2$ and $A3$ with probabilities x_1 , x_2 and x_3 respectively with $0 \leq x_i \leq 1$ while Player B mixes strategies $B1$, $B2$ and $B3$ with probabilities y_1 , y_2 and y_3 respectively with $0 \leq y_i \leq 1$. Player A's linear (Primal) program is stated as

$$\begin{aligned} \max \quad & z = v \\ s.t \quad & v - 3x_1 + 2x_2 + 5x_3 \leq 0 \\ & v + x_1 - 4x_2 + 6x_3 \leq 0 \\ & v + 3x_1 + x_2 - 2x_3 \leq 0 \\ & x_1 + x_2 + x_3 = 1 \\ & x_1,\ x_2,\ x_3 \geq 0 \\ & v \ \ unrestricted \end{aligned}$$

The LPP formulation for the Primal program is presented as

$$\max C^T x \quad s.t \ \ Ax + s = \alpha; \qquad (66)$$

where $x \geq 0$, $s \geq 0$ and the constructed matrix operator A and other coefficient vectors are stated below.

$$A = \begin{bmatrix} 1 & | & -3 & 2 & 5 \\ 1 & | & 1 & -4 & 6 \\ 1 & | & 3 & 1 & -2 \\ 0 & | & 1 & 1 & 1 \end{bmatrix} \quad x = \begin{bmatrix} v \\ x_1 \\ x_2 \\ x_3 \end{bmatrix},$$

$$, \ C = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad and \quad \alpha = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}.$$

Player B's linear (Dual) program is stated as

$$\begin{aligned} & \min w = v \\ & s.t v - 3y_1 + y_2 + 3y_3 \geq 0 \\ & v - 2y_1 - 4y_2 + y_3 \geq 0 \\ & v + 5y_1 + 6y_2 - 2y_3 \geq 0 \\ & y_1 + y_2 + y_3 = 1 \\ & y_1, y_2, y_3 \geq 0 \\ & v \ \ unrestricted \end{aligned}$$

While the LPP formulation for the Dual program is presented as

$$\min \ D^T y \quad s.t \ \ By - s = \overline{\alpha}; \qquad (67)$$

where $y \geq 0$, $s \geq 0$, and the constructed matrix operator B and other coefficient vectors are stated below.

$$B = \begin{bmatrix} 1 & | & -3 & 1 & 3 \\ 1 & | & 2 & -4 & 1 \\ 1 & | & 5 & 6 & -2 \\ 0 & | & 1 & 1 & 1 \end{bmatrix}, \ y = \begin{bmatrix} v \\ y_1 \\ y_2 \\ y_3 \end{bmatrix},$$

$$s = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ 0 \end{bmatrix}, \ D = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad and \quad \overline{\alpha} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}.$$

Applying the constructed ADMM Algorithm above to the LPP in example 2 yields the following Primal and Dual probabilities, residues and convergences indicated in figures 5, 6, 7 and 8 below.

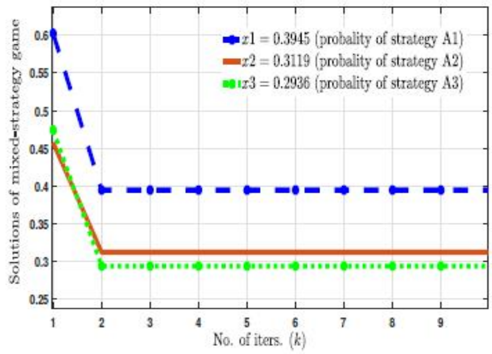


Figure 5: Probabilities for the mixed strategies of Player A.

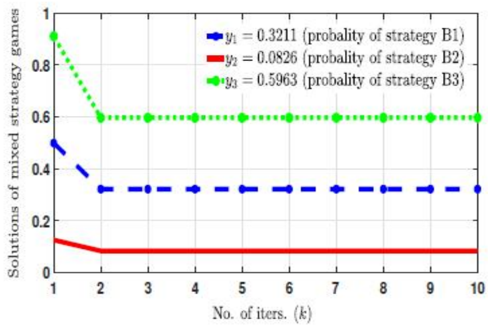
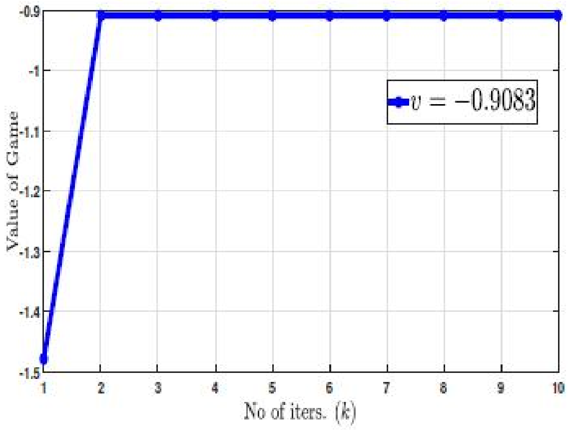


Figure 6: Probabilities for the mixed strategies of Player B.

Bima Journal of Science and Technology, Vol. 9(1A) Mar, 2025 ISSN: 2536-6041

DOI: 10.56892/bima.v9i1A.1258



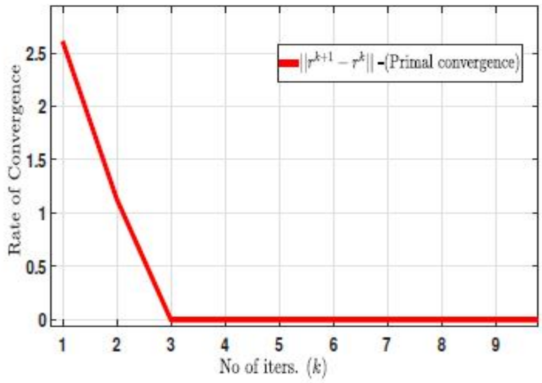


Figure 7: Value of Game.

RESULTS AND DISCUSSION

The implementation of the ADMM on example 1, of the general LPP, yields the results $x = 2.6832$ and $u = 2.5991$ and the objective value $v = 8.2874$. The primal and dual residuals approach convergence at about 10 iterations. The results for the primal program of the game problem of example 2 are given $as\, x1 = 0.3945, x2 = 0.3119$ and $x3 = 0.2936$ expressed in dashed, regular and dotted lines in figure 5 above for strategies A1, A2 and A3 respectively, after 3 iterations, while $y1 = 0.3211, y2 = 0.0826$ and $y3 = 0.5963$ are the results for the dual program for strategies B1, B2 and B3 expressed in the dashed, regular and dotted lines respectively in figure 6 above. The value of the game is given as $v = -0.9083$ and indicated in figure 7. The rate of convergence of the accelerated ADMM, after 3 iterations, is indicated in figure 8.

Funding
This work was funded solely by the Author.

CONCLUSION

In this research paper, we derived the Proximal Point Algorithm (ADMM) for the classical Linear Programming problem (LPP) with application to the Games with no saddle point. The algorithm was implemented on both the primal and dual of the LPP with the use of an accelerator variant in the sense of Nesterov to speed up the rate of convergence and its performance. The rate of convergence of the LPP, for the game theory, resulting from the proximal point Algorithm (ADMM) were analyzed and the results achieved with modest accuracy in only a few iterations. The Transportation model can also be structured in a manner amenable to the proximal point algorithm provided the constructed matrix operator is invertible.

REFERENCES

Boyd, S., Parikh, N., Chu, E., Peleato, B and Eckstein, J. (2011), *Distributed Optimization and Statistical learning via the Alternating Direction Method of Multipliers*, Foundations and trends in Machine Learning. 3, 11-22.

Dawodu, K. A. (2021), *Extension of ADMM Algorithm in Solving Optimal Control Model Governed by Partial Differential Equation*, Nig. Soc. Phy. Sci. 3(2021) 105-115.

Dantzig, G., (1963) *Linear Programming and Extensions*, Princeton University Press, Princeton, NJ. Eckstein, J. and Bertsekas,

DOI: 10.56892/bima.v8i3B.842

Famaey, J., De Lathauwer, L., & Lee, K. S. (2014). *Application of ADMM to the LASSO problem.* IEEE Transactions on Signal Processing, 62(20), 5114-5123.

Ghadimi, E. Teixeira, A., Shames, I. and Johansson, M. (2014), *Optimal Parameter Selection for the Alternating Direction Method of Multipliers (ADMM):* Quadratic Problems, Linnaeus center, Electrical Engineering, Royal Institute of Technology, Sidney.

Guo, D., Li, Y., Zhang, H., & Wang, W. (2019). *A parallel ADMM algorithm specifically designed for linear programming problems with a large number of constraints*. Journal of Parallel and Distributed Computing, 132, 37-47.

Hong, M., & Sun, D. (2016). *Implementation of Primal-Dual Methods, including ADMM, for solving large-scale linear programming problems.* Journal of Computational Mathematics, 34(4),427-442.

Nesterov, Y. E. (1983), *A method for Solving the Convex Programming Problem with Convergence Rate O(1/k 2)*. Dokl. Akad. Nauk SSSR, 269, 543-547.

Patrascu, A., & Rosu, M. G. (2019). *An ADMM-based approach for solving linear programming problems with both equality and inequality constraints.* Journal of Optimization Theory and Applications, 182(1), 235-254.

Qin, J., Yuan, X., & Wang, R. (2017). *A new Primal-Dual method for large-scale linear programming with linear inequality constraints.* Journal of Optimization Theory and Applications, 174(2), 589-606.

Yang, Y., & Zhang, J. (2017). *An alternating direction method for solving semidefinite programming problems with linear constraints.* Journal of Optimization Theory and Applications, 174(1), 271-287

Yuan, X., & Wang, R. (2017). *An accelerated ADMM for linear and convex quadratic network utility maximization*. IEEE Transactions on Signal Processing, 65(16), 4240-4253.