



UNIVERSITY OF NOVI SAD
FACULTY OF SCIENCES

DEPARTMENT OF
MATHEMATICS AND
INFORMATICS



IMPLEMENTATION OF INTERIOR-POINT METHOD FOR SOLVING LINEAR PROGRAMING OPTIMIZATION PROBLEM IN FORM OF GRAPHICAL MODEL AND BELIEF PROPAGATION ALGORITHM

MASTER THESIS

Author: Ivan Hornjak
Mentor: Dejan Vukobratović

Acknowledgement

I would like to express my sincere gratitude to my supervisor, Prof. Dejan Vukobratović, for his guidance, support, and especially for his patience throughout the completion of this thesis.

I am deeply grateful to my family for their unconditional support, patience and understanding throughout what turned out to be a much longer academic journey than originally planned, and to my friends for all their support and help along the way.

Finally, I would like to express my special gratitude to my wife and daughter, whose love, patience, and constant support have been invaluable, especially during the final stages of my studies and the completion of this thesis.

Abstract

Interior-point methods represent one of the most efficient approaches for solving large-scale linear programming problems. However, their main computational bottleneck lies in solving the linear systems that arise at each Newton iteration, which typically requires matrix factorization or implicit matrix inversion. For large and sparse problems, this step can become computationally expensive and difficult to parallelize.

In the work of Bickson et al. [\[1\]](#), an alternative framework was proposed in which the Newton system arising in linear programming is reinterpreted as a probabilistic inference problem and solved using Gaussian Belief Propagation (GaBP). This approach establishes a connection between interior-point optimization, numerical linear algebra, and probabilistic graphical models.

The objective of this thesis is to provide a systematic analysis and exposition of this framework. Starting from the logarithmic barrier formulation of linear programming, the thesis derives the associated Newton system and shows how it can be transformed into a least-squares problem and subsequently into a Maximum A Posteriori (MAP) inference problem in a Gaussian graphical model. The Gaussian Belief Propagation algorithm is then formulated as a solver for the resulting system, and its convergence properties are discussed. Furthermore, the methodology is extended to the primal-dual interior-point formulation, and an illustrative numerical example is presented.

The thesis demonstrates how the Newton step of an interior-point method can be interpreted as probabilistic inference in a graphical model and provides a unified perspective on the relationship between convex optimization, numerical linear algebra, and probabilistic graphical models.

Contents

<i>Acknowledgement</i>	2
<i>Abstract</i>	3
<i>Contents</i>	4
1. Introduction	6
1.1 Motivation and Significance of the Thesis	6
1.2 Challenges of Interior-Point Methods	7
1.3 Transition from Linear Algebra to Graphical Models.....	8
1.4 Objectives and Contributions of the Thesis.....	9
2. Linear Programming and the Log-Barrier Method	11
2.1 Standard Form of the Linear Programming Problem.....	11
2.2 Log-Barrier Method.....	12
2.3 Newton's Method for the Barrier Problem.....	14
2.4 Reduction to a Least-Squares Problem	15
3. Probabilistic Interpretation of the Least-Squares Problem	18
3.1 Least-Squares as Quadratic Optimization.....	18
3.2 Quadratic Forms and Gaussian Distributions	19
3.3 Construction of the Gaussian Graphical Model.....	21
4. Gaussian Belief Propagation	24
4.1 Overview of Belief Propagation	24
4.2 Belief Propagation in Gaussian Models	26
4.3 Message Update Equations in Information Form.....	27
4.4 Convergence Properties of Gaussian Belief Propagation.....	29
5. GaBP-Based Interior-Point Method	31
5.1 Primal-Dual Interior-Point Formulation	31

5.2	Newton System in Primal-Dual Form	32
5.3	GaBP Solver for the Newton Step	34
5.4	Complete BP-Based LP Algorithm	36
5.5	Computational Considerations	39
6.	<i>Convergence Analysis</i>	42
6.1	Convergence of Gaussian Belief Propagation.....	42
6.2	Exactness of the Fixed Point.....	43
6.3	Convergence within the Primal-Dual Framework.....	45
7.	<i>Numerical Example</i>	47
7.1	Definition of an Illustrative Linear Programming Problem	47
7.2	Construction of the Primal-Dual Newton System	48
7.3	Gaussian Interpretation and GaBP Computation	51
7.4	Recovery of the Newton Directions.....	52
8.	<i>Conclusion</i>	55
	<i>References</i>	57
	<i>Biography</i>	58

1. Introduction

1.1 Motivation and Significance of the Thesis

Linear programming (LP) represents one of the fundamental areas of mathematical optimization and has wide applications in engineering, economics, operations research, telecommunications, machine learning, and logistics. A standard linear programming problem consists of minimizing (or maximizing) a linear objective function subject to a set of linear equality and inequality constraints. Due to its convex nature, an LP problem admits a globally optimal solution whenever the feasible set is nonempty.

Interior-point methods, especially primal-dual variants, are now considered a standard approach for solving large and sparse LP problems. Their primary advantage lies in their fast convergence and numerical stability. However, at each iteration of these methods, it is necessary to solve a system of linear equations arising from the Newton step. Solving this system often constitutes the most computationally expensive part of the algorithm, particularly when the problem dimension is large.

In the classical approach, this linear system is solved using direct methods from numerical linear algebra, such as Cholesky factorization or LU decomposition. These methods require the computation, or implicit inversion, of the Hessian matrix (or its reduced forms), which can be computationally demanding, especially in distributed or parallel computing environments.

Modern trends in optimization emphasize scalable and distributed algorithms capable of exploiting the underlying structure of the problem, particularly sparsity. In this context, it is natural to ask whether the classical linear-algebraic approach can be reinterpreted within a different formal framework that enables more efficient and potentially distributed computation.

One such approach is based on probabilistic graphical models and the Belief Propagation (BP) algorithm. In particular, Gaussian Belief Propagation (GaBP) enables the solution of linear systems through local message passing between nodes in a graph, without explicit matrix inversion. In this way, the problem is shifted from deterministic linear algebra to probabilistic inference.

This thesis investigates precisely this connection: how the interior-point method for solving linear programming problems can be transformed into a graphical model and solved using the Belief Propagation algorithm. This transformation opens the possibility of developing a distributed and scalable LP solver based on principles of probabilistic inference.

The connection between linear programming and Gaussian Belief Propagation was originally proposed by Bickson et al. [1], who demonstrated that the Newton systems arising in interior-point methods can be reformulated as Gaussian inference problems and solved using distributed message passing. Their work provided one of the first explicit links between convex optimization and probabilistic graphical models in the context of linear programming.

1.2 Challenges of Interior-Point Methods

Interior-point methods represent a class of algorithms for solving convex optimization problems, including linear programming, that approach the optimal solution by moving through the interior of the feasible set. Interior-point methods generate a sequence of iterates that remain strictly within the region defined by the inequality constraints.

The fundamental idea behind interior-point methods is based on transforming the original constrained problem into a sequence of smooth approximations using barrier functions. In the context of linear programming, the logarithmic barrier function is most commonly used, as it penalizes approaching the boundary of the feasible set. In this way, the nonnegativity constraints on the variables are replaced by a penalization term added to the objective function, resulting in a problem without explicit inequality constraints but with a modified objective.

The resulting problem is typically solved using Newton’s method, which leads to a system of linear equations that must be solved at each iteration. In the primal-dual formulation of interior-point methods, this system arises from the Karush–Kuhn–Tucker (KKT) optimality conditions and involves block matrices that couple the primal and dual variables.

A key numerical characteristic of interior-point methods is their polynomial-time complexity and fast (often quadratic) local convergence of the Newton step. In practice, the number of iterations is relatively small and often nearly independent of the problem dimension.

However, each iteration requires solving a large system of linear equations, whose size depends on the number of variables and constraints.

The primary computational burden of interior-point methods lies precisely in solving this linear system. Although the matrix structure, particularly sparsity, can be exploited, direct factorization methods may lead to an increase in matrix density during the elimination process (the so-called fill-in effect), thereby increasing memory requirements and computational cost.

An additional challenge arises from the presence of the Hessian matrix of the barrier function in the Newton step. Although this Hessian is diagonal, its interaction with the constraints results in a reduced system that requires inversion or factorization of matrices. Direct methods for solving linear systems, such as Cholesky factorization, are numerically stable and widely used, but their computational complexity grows approximately cubically with the system dimension in the worst case. For large-scale problems, particularly those involving millions of variables and constraints, such approaches become impractical without sophisticated techniques for exploiting matrix structure.

These considerations motivate the search for alternative approaches that enable solving the Newton system without explicit matrix inversion or global factorization. The idea explored in this thesis is based on reinterpreting the linear system as a probabilistic inference problem over a Gaussian graphical model. In this framework, the system can be solved iteratively and locally using the Belief Propagation algorithm, thereby enabling a potentially distributed and scalable implementation of the interior-point method.

1.3 Transition from Linear Algebra to Graphical Models

The classical approach to solving the linear systems arising in interior-point methods relies on techniques from numerical linear algebra, such as matrix factorizations or iterative solvers. In this setting, the system is treated purely as an algebraic object defined by a coefficient matrix and a right-hand-side vector. The solution is obtained through matrix manipulation, without explicitly modeling the structural dependencies among individual variables.

However, any linear system can be equivalently expressed as the minimization of a quadratic function, i.e., as a least-squares problem. A quadratic form, in turn, can be interpreted as the negative logarithm of a Gaussian probability density function. Consequently, solving a system of linear equations becomes equivalent to performing a Maximum A Posteriori (MAP) estimation problem in a Gaussian probabilistic model.

This reinterpretation enables a transition from deterministic linear algebra to probabilistic graphical models. In such a model, variables correspond to nodes in a graph, and dependencies are encoded through edges associated with the nonzero entries of the system matrix. The sparsity structure of the matrix directly determines the topology of the graphical model.

Within Gaussian graphical models, MAP inference can be efficiently performed using the GaBP algorithm. Instead of explicitly inverting matrices, GaBP computes the solution through iterative local message passing between neighboring nodes. If convergence is achieved, the resulting marginal means coincide exactly with the solution of the original linear system.

1.4 Objectives and Contributions of the Thesis

The primary objective of this thesis is to analyze, reconstruct, and systematically present the connection between interior-point methods and probabilistic graphical models originally proposed in [1]. Particular emphasis is placed on the transformation of the Newton system into a Gaussian inference problem and on the role of Gaussian Belief Propagation as an alternative computational mechanism for solving the resulting linear systems.

Specifically, the thesis aims to:

- derive the Newton system arising from the logarithmic barrier formulation of linear programming
- reduce this system to a least-squares formulation
- demonstrate the equivalence between least-squares optimization and MAP inference in a Gaussian graphical model
- construct the corresponding graphical model in information form

- formulate and analyze the Gaussian Belief Propagation algorithm as a solver for the Newton step
- extend the construction to the primal-dual interior-point method
- analyze convergence conditions and provide theoretical insight into the stability of the GaBP-based approach.

The contribution of this thesis lies in providing a unified and self-contained presentation of the theoretical framework connecting linear programming, interior-point methods, Gaussian graphical models, and Belief Propagation. By reconstructing the derivation from first principles and analyzing the resulting probabilistic interpretation, the thesis offers a detailed exposition of the methodology originally proposed in the literature and highlights its conceptual significance and computational implications.

2. Linear Programming and the Log-Barrier Method

2.1 Standard Form of the Linear Programming Problem

Linear programming problems can be formulated in several equivalent forms. In this thesis, we consider the standard equality-constrained form with nonnegativity constraints, which is particularly suitable for the derivation of interior-point methods.

The standard linear programming problem is defined as:

$$\begin{aligned} \min_{x \in \mathbb{R}^n} c^T x \\ \text{subject to } Ax = b, x \geq 0 \end{aligned}$$

Where:

- $x \in \mathbb{R}^n$ is the vector of decision variables,
- $c \in \mathbb{R}^n$ is the cost (objective) vector,
- $A \in \mathbb{R}^{m \times n}$ is the constraint matrix,
- $b \in \mathbb{R}^m$ is the constraint vector.

We assume that:

1. The matrix A has full row rank, i.e., $\text{rank}(A) = m < n$ which ensures that the equality constraints are linearly independent.
2. The feasible set is nonempty.
3. The problem is strictly feasible, meaning that there exists a vector x such that: $Ax = b, x > 0$.

The strict feasibility assumption is important for the applicability of the logarithmic barrier method and ensures the existence of an interior trajectory (central path).

The feasible set $\{x \in \mathbb{R}^n | Ax = b, x \geq 0\}$ defines a convex polyhedron. Since both the objective function and the feasible set are convex, any local minimum is also a global minimum.

The optimal solution of a linear program, when it exists and is bounded, lies at an extreme point (vertex) of the feasible polyhedron. However, interior-point methods do not traverse vertices; instead, they approach the solution through the interior of the feasible region.

The main difficulty in solving the standard LP formulation using Newton-type methods lies in handling the inequality constraint $x \geq 0$. Since Newton's method applies to unconstrained smooth optimization problems, the inequality constraints must be incorporated into the objective function. This leads naturally to the logarithmic barrier reformulation, which will be introduced in the next section.

2.2 Log-Barrier Method

We begin with the standard LP problem and its logarithmic barrier reformulation, following the treatment presented in [\[2\]](#):

$$\begin{aligned} \min_{x \in \mathbb{R}^n} c^T x \\ \text{subject to } Ax = b, x \geq 0. \end{aligned}$$

The inequality constraint $x \geq 0$ is replaced by a barrier function defined as

$$\phi(x) = - \sum_{i=1}^n \log x_i$$

The barrier function is defined only for $x_i > 0$ and satisfies

$$\phi(x) \rightarrow +\infty \text{ as } x_i \rightarrow 0^+.$$

This property ensures that the iterates remain strictly within the interior of the feasible region.

The barrier problem is then defined as

$$\min_x c^T x - \mu \sum_{i=1}^n \log x_i$$

subject to $Ax = b$,

where $\mu > 0$ is the barrier parameter.

For each value of $\mu > 0$, we define the solution

$$x(\mu) = \arg \min_{Ax=b, x>0} \left(c^T x - \mu \sum_{i=1}^n \log x_i \right)$$

The set of points $\{x(\mu)\}_{\mu>0}$ is called the central path.

Under standard assumptions (strict feasibility), it holds that

$$\lim_{\mu \rightarrow 0} x(\mu) = x^*$$

where x^* is the optimal solution of the original linear programming problem.

Since the barrier problem contains only equality constraints, we form the Lagrangian:

$$\mathcal{L}(x, y) = c^T x - \mu \sum_{i=1}^n \log x_i + y^T (Ax - b)$$

where $y \in \mathbb{R}^m$ is the vector of Lagrange multipliers.

Optimality requires that the gradient of the Lagrangian with respect to both x and y be equal to zero.

Taking the derivative with respect to x :

$$\nabla_x \mathcal{L} = c - \mu X^{-1} \mathbf{1} + A^T y = 0,$$

where $X = \text{diag}(x_1, \dots, x_n)$ and $\mathbf{1}$ is the all-ones vector.

Taking the derivative with respect to y :

$$\nabla_y \mathcal{L} = Ax - b = 0.$$

We obtain the nonlinear system

$$\begin{cases} Ax = b \\ A^T y + c - \mu X^{-1} \mathbf{1} = 0 \end{cases}$$

This system characterizes the central path.

2.3 Newton's Method for the Barrier Problem

In order to solve the logarithmic barrier problem

$$\begin{aligned} \min_x f(x) &= c^T x - \mu \sum_{i=1}^n \log x_i \\ &\text{subject to } Ax = b, \end{aligned}$$

we apply Newton's method to compute a search direction within equality-constrained feasible set.

Let $\tilde{x}$ denote the current feasible iterate satisfying $A\tilde{x} = b$ and $\tilde{x} > 0$. Newton's method is based on a second-order Taylor approximation of the objective around $\tilde{x}$:

$$f(\tilde{x} + \Delta x) \approx f(\tilde{x}) + \nabla f(\tilde{x})^T \Delta x + \frac{1}{2} \Delta x^T \nabla^2 f(\tilde{x}) \Delta x$$

The Newton search direction Δx is obtained by minimizing this quadratic approximation subject to maintaining feasibility with respect to the equality constraints.

The gradient of the barrier objective is:

$$\nabla f(x) = c - \mu X^{-1} \mathbf{1}$$

The Hessian is diagonal and given by:

$$\nabla^2 f(x) = \mu X^{-2}$$

These expressions are standard in logarithmic barrier methods and follow directly from derivatives of $-\log x_i$.

Because the iterate must remain feasible with respect to the equality constraints, the Newton step must satisfy:

$$A(\tilde{x} + \Delta x) = b$$

Since $A\tilde{x} = b$, this implies the linear constraint on the search direction: $A\Delta x = 0$.

Therefore, Δx is computed as the solution of the following equality-constrained quadratic problem:

$$\min_{\Delta x} \nabla f(\tilde{x})^T \Delta x + \frac{1}{2} \Delta x^T \nabla^2 f(\tilde{x}) \Delta x$$

subject to $A\Delta x = 0$.

Introducing a Lagrange multiplier y for the constraint $A\Delta x = 0$, the optimality conditions become:

$$\begin{aligned} \nabla^2 f(\tilde{x}) \Delta x + \nabla f(\tilde{x}) - A^T y &= 0 \\ A\Delta x &= 0 \end{aligned}$$

Substituting $\nabla f(\tilde{x})$ and $\nabla^2 f(\tilde{x})$ yields:

$$\begin{aligned} \mu X^{-2} \Delta x + c - \mu X^{-1} \mathbf{1} - A^T y &= 0 \\ A\Delta x &= 0. \end{aligned}$$

This is the Newton system associated with the barrier formulation.

The key observation is that the above conditions can be manipulated to eliminate Δx and obtain a symmetric linear system in y . This system takes the form of normal equations and can be recognized as arising from a linear least-squares problem.

2.4 Reduction to a Least-Squares Problem

Newton system associated with the barrier problem:

$$\begin{aligned} \mu X^{-2} \Delta x + c - \mu X^{-1} \mathbf{1} - A^T y &= 0 \\ A\Delta x &= 0. \end{aligned}$$

The goal is to eliminate Δx and obtain a linear system involving only the variable y .

The first equation can be rearranged as:

$$\mu X^{-2} \Delta x = A^T y - c + \mu X^{-1} \mathbf{1}.$$

Since X^{-2} is a diagonal matrix, it is easily invertible. Multiplying both sides by X^{-2} gives:

$$\mu \Delta x = X^2 A^T y - X^2 c + \mu X \mathbf{1}.$$

Because $X \mathbf{1} = x$, we obtain:

$$\mu \Delta x = X^2 A^T y - X^2 c + \mu x$$

Thus,

$$\Delta x = \frac{1}{\mu} (X^2 A^T y - X^2 c + \mu x)$$

Recall that the Newton direction must satisfy the feasibility condition:

$$A\Delta x = 0$$

Substituting the expression for Δx , we obtain:

$$A\left(\frac{1}{\mu} (X^2 A^T y - X^2 c + \mu x)\right) = 0$$

Multiplying by μ yields:

$$AX^2 A^T y - AX^2 c + \mu Ax = 0.$$

Since the current iterate is feasible, we have:

$$Ax = b$$

Therefore,

$$AX^2 A^T y - AX^2 c + \mu b = 0$$

or equivalently,

$$AX^2 A^T y = AX^2 c - \mu b$$

This is a linear system in the variable y .

The matrix

$$M = AX^2 A^T$$

is:

- symmetric,
- positive definite (since X^2 is a positive definite diagonal matrix and A has full row rank).

Hence, the system is well defined and admits a unique solution.

The system

$$AX^2 A^T y = AX^2 c - \mu b$$

can be recognized as a system of normal equations arising from a least-squares problem.

Define

$$F = XA^T.$$

Then

$$F^T F = AX^2 A^T.$$

The right-hand side can be expressed accordingly so that the system takes the standard normal-equations form:

$$F^T F y = F^T g,$$

which corresponds to the least-squares problem:

$$\min_y \| Fy - g \|^2.$$

Therefore, the Newton step of the interior-point method reduces to solving a weighted least-squares problem.

3. Probabilistic Interpretation of the Least-Squares Problem

3.1 Least-Squares as Quadratic Optimization

It is showed that the Newton step of the interior-point method reduces to solving a least-squares problem of the form

$$\min_y \|Fy - g\|^2,$$

where $F = XA^T$ and g is a vector determined by the current iteration.

The squared Euclidean norm can be written as

$$\|Fy - g\|^2 = (Fy - g)^T (Fy - g).$$

Expanding the expression yields

$$\|Fy - g\|^2 = y^T F^T F y - 2g^T F y + g^T g.$$

Since the term $g^T g$ does not depend on the variable y , it does not affect the minimization and may be omitted when determining the optimal solution.

Therefore, the least-squares problem is equivalent to minimizing the quadratic function

$$\min_y \left(\frac{1}{2} y^T (2F^T F) y - 2g^T F y \right).$$

This is commonly written in standard quadratic form as

$$\min_y \left(\frac{1}{2} y^T H y - r^T y \right),$$

where $H = F^T F, r = F^T g$.

The first-order optimality condition is obtained by setting the gradient to zero:

$$\nabla_y \left(\frac{1}{2} y^T H y - r^T y \right) = Hy - r = 0.$$

Hence, the optimal solution satisfies

$$Hy = r,$$

which are precisely the normal equations associated with the least-squares problem.

Since $H = F^T F$, the matrix H is:

- symmetric,
- positive semidefinite,
- positive definite if F has full column rank.

Due to the structure $F = XA^T$ and the full-rank assumption on the constraint matrix, the system is positive definite and the solution is unique.

Therefore obtained is an optimization problem of the form

$$\min_y \left(\frac{1}{2} y^T H y - r^T y \right),$$

which is:

- a strictly convex quadratic optimization problem,
- fully equivalent to solving a system of linear equations,
- structurally identical to the energy function of a Gaussian model.

3.2 Quadratic Forms and Gaussian Distributions

In this section, it is demonstrated that a quadratic form

$$\min_y \left(\frac{1}{2} y^T H y - r^T y \right)$$

can be interpreted as the negative logarithm of a Gaussian probability density function. This observation establishes the connection between deterministic optimization and probabilistic inference.

A multivariate Gaussian distribution over a random vector $y \in \mathbb{R}^m$ can be written in its standard form as

$$p(y) = \frac{1}{(2\pi)^{m/2} |\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(y - \mu)^T \Sigma^{-1}(y - \mu)\right).$$

where:

- μ is the mean vector,
- Σ is the covariance matrix,
- Σ^{-1} is the precision matrix.

For purposes in this thesis, it is more convenient to use the information form of the Gaussian distribution. In this form, the density is written as

$$p(y) \propto \exp\left(-\frac{1}{2}y^T H y + r^T y\right),$$

where:

- $H = \Sigma^{-1}$ is the precision matrix,
- $r = \Sigma^{-1}\mu$ is the information vector.

This representation omits the normalization constant, which is irrelevant for optimization and MAP estimation.

Taking the negative logarithm of the Gaussian density gives

$$-\log p(y) = \frac{1}{2}y^T H y - r^T y.$$

This expression is identical to the quadratic objective function derived in Section 3.1.

Therefore, minimizing the quadratic function

$$\frac{1}{2}y^T H y - r^T y$$

is equivalent to maximizing the Gaussian density $p(y)$.

The minimizer of the quadratic objective

$$y^* = \arg \min_y \left(\frac{1}{2} y^T H y - r^T y \right)$$

is therefore equivalent to the Maximum A Posteriori (MAP) estimate of the Gaussian distribution

$$y^* = \arg \max_y p(y).$$

From the first-order optimality condition

$$H y^* = r,$$

we obtain

$$y^* = H^{-1} r.$$

Thus, solving the linear system

$$H y = r$$

is equivalent to computing the MAP estimate of a Gaussian graphical model in information form.

3.3 Construction of the Gaussian Graphical Model

We now construct the corresponding Gaussian graphical model and describe its structural properties.

A fundamental property of Gaussian distributions is that the sparsity pattern of the precision matrix H determines conditional independencies between variables.

Specifically, $H_{ij} = 0 \iff y_i$ and y_j are conditionally independent given the remaining variables. This property defines a Gaussian Markov Random Field (GMRF).

The graphical model is constructed as follows:

- Each variable y_i corresponds to a node in the graph.
- An undirected edge is placed between nodes i and j if and only if $H_{ij} \neq 0$, for $i \neq j$.

Thus, the topology of the graph is directly determined by the sparsity structure of the precision matrix.

In our case,

$$H = F^T F,$$

where $F = XA^T$, meaning that the graph structure ultimately reflects the structure of the original constraint matrix of the linear program. If the original matrix is sparse, the resulting graphical model is also sparse.

The Gaussian density in information form can be expanded as

$$y^T H y = \sum_i H_{ii} y_i^2 + 2 \sum_{i < j} H_{ij} y_i y_j.$$

Therefore, the joint distribution can be written as a product of local and pairwise potentials:

$$p(y) \propto \prod_i \psi_i(y_i) \prod_{(i,j) \in E} \psi_{i,j}(y_i, y_j)$$

where

$$\begin{aligned} \psi_i(y_i) &= \exp\left(-\frac{1}{2} H_{ii} y_i^2 + r_i y_i\right), \\ \psi_{i,j}(y_i, y_j) &= \exp(-H_{ij} y_i y_j). \end{aligned}$$

This factorization corresponds to a pairwise Gaussian Markov Random Field.

Several important observations follow:

1. The degree of each node equals the number of nonzero off-diagonal entries in the corresponding row of H .
2. Sparsity of H implies sparse connectivity of the graph.
3. All interactions are pairwise and quadratic, making the model fully Gaussian.

The solution of the linear system can therefore be obtained by performing inference over this graphical model. Instead of explicitly computing H^{-1} , one can compute marginal means via message passing, provided that the inference algorithm converges.

4. Gaussian Belief Propagation

4.1 Overview of Belief Propagation

Belief Propagation (BP) is an algorithm for performing inference on graphical models [\[3\]](#). It computes marginal distributions of variables by passing local messages between nodes of a graph. Originally developed for tree-structured graphs, BP provides exact inference when the underlying graphical model contains no cycles. When applied to graphs with cycles, the algorithm becomes iterative and approximate, though in many cases it still converges to useful solutions.

A graphical model represents a joint probability distribution over a set of random variables using a graph structure. There are two main types:

- Bayesian Networks (directed graphs)
- Markov Random Fields (undirected graphs)

In this thesis, we focus on undirected graphical models, specifically Gaussian Markov Random Fields (GMRFs). In an undirected graph:

- Nodes represent random variables.
- Edges represent statistical dependencies.
- The joint distribution factorizes over cliques of the graph.

For a pairwise Markov Random Field, the joint distribution can be written as

$$p(x) \propto \prod_i \psi_i(x_i) \prod_{(i,j) \in E} \psi_{ij}(x_i, x_j).$$

where:

- $\psi_i(x_i)$ are node potentials,
- $\psi_{ij}(x_i, x_j)$ are pairwise potentials.

The goal of inference is to compute marginal distributions $p(x_i)$ or in the case of MAP inference, to compute $\arg \max_x p(x)$.

Belief Propagation computes marginals by iteratively sending messages along edges. The message from node i to node j is denoted:

$$m_{i \rightarrow j}(x_j).$$

It is computed using all incoming messages to node i , except the one from j . For discrete variables, the message update rule is

$$m_{i \rightarrow j}(x_j) = \sum_{x_i} \psi_{ij}(x_i, x_j) \psi_i(x_i) \prod_{k \in N(i) \setminus j} m_{k \rightarrow i}(x_i),$$

where $N(i)$ denotes the neighbors of node i .

After convergence, the belief at node i is computed as

$$b_i(x_i) \propto \psi_i(x_i) \prod_{k \in N(i)} m_{k \rightarrow i}(x_i).$$

On tree-structured graphs, these beliefs coincide exactly with true marginals. On graphs with cycles, convergence is not guaranteed, but when it occurs, the beliefs often provide accurate approximations.

In our case:

- The variables are continuous.
- The potentials are Gaussian.
- The graphical model corresponds to a Gaussian Markov Random Field.

This leads to a specialized version of BP known as Gaussian Belief Propagation, which operates entirely with Gaussian messages.

4.2 Belief Propagation in Gaussian Models

In this section, we specialize the general Belief Propagation framework to Gaussian graphical models. When all node and edge potentials are Gaussian, the resulting algorithm is known as Gaussian Belief Propagation (GaBP). Because Gaussian distributions are closed under multiplication and marginalization, all messages and beliefs remain Gaussian throughout the algorithm.

In Chapter 3 we expressed the distribution as a product of local and pairwise Gaussian potentials:

$$p(y) \propto \prod_i \psi_i(y_i) \prod_{(i,j) \in E} \psi_{ij}(y_i, y_j),$$

where:

$$\begin{aligned} \psi_i(y_i) &= \exp\left(-\frac{1}{2}H_{ii}y_i^2 + r_i y_i\right), \\ \psi_{ij}(y_i, y_j) &= \exp(-H_{ij}y_i y_j). \end{aligned}$$

In Gaussian BP, every message is itself a Gaussian function.

The message sent from node i to node j has the general form

$$m_{i \rightarrow j}(y_j) \propto \exp\left(-\frac{1}{2}\alpha_{ij}y_j^2 + \beta_{ij}y_j\right),$$

where

- α_{ij} is the precision term,
- β_{ij} is the information term.

Thus, each message can be represented compactly by two scalar parameters $(\alpha_{ij}, \beta_{ij})$.

The belief at node i is obtained by multiplying the local potential and all incoming messages:

$$b_i(y_i) \propto \psi_i(y_i) \prod_{k \in N(i)} m_{k \rightarrow i}(y_i).$$

Since all terms are Gaussian, the product is also Gaussian:

$$b_i(y_i) \propto \exp\left(-\frac{1}{2}\hat{H}_i y_i^2 + \hat{r}_i y_i\right),$$

where:

$$\begin{aligned}\hat{H}_i &= H_{ii} + \sum_{k \in N(i)} \alpha_{ki}, \\ \hat{r}_i &= r_i + \sum_{k \in N(i)} \beta_{ki}.\end{aligned}$$

The marginal mean estimate is therefore

$$\hat{y}_i = \frac{\hat{r}_i}{\hat{H}_i}.$$

The key property that makes GaBP possible is that:

- Multiplication of Gaussian functions yields another Gaussian.
- Marginalization of a Gaussian over one variable yields another Gaussian.

Therefore, the message update step reduces to simple algebraic updates of precision and information parameters. This dramatically simplifies inference compared to discrete BP.

4.3 Message Update Equations in Information Form

In this section, we derive the explicit update equations for Gaussian Belief Propagation in information form.

Each message from node i to node j is Gaussian:

$$m_{i \rightarrow j}(y_j) \propto \exp\left(-\frac{1}{2}\alpha_{ij}y_j^2 + \beta_{ij}y_j\right).$$

Before sending a message to node j , node i aggregates all incoming information except from j .

Define

$$\begin{aligned}\tilde{H}_i^{\setminus j} &= H_{ii} + \sum_{k \in N(i) \setminus j} \alpha_{ki}, \\ \tilde{r}_i^{\setminus j} &= r_i + \sum_{k \in N(i) \setminus j} \beta_{ki}.\end{aligned}$$

These represent the effective precision and information at node i excluding the contribution from node j .

To compute the message from i to j , we marginalize over y_i the product of:

- the local potential at node i ,
- incoming messages (excluding j),
- the pairwise interaction between i and j .

This yields the update equations

$$\begin{aligned}\alpha_{ij} &= -\frac{H_{ij}^2}{\tilde{H}_i^{\setminus j}}, \\ \beta_{ij} &= -\frac{H_{ij}\tilde{r}_i^{\setminus j}}{\tilde{H}_i^{\setminus j}}.\end{aligned}$$

These equations define the GaBP message updates. After messages converge, the belief at node i is given by

$$b_i(y_i) \propto \exp\left(-\frac{1}{2}\hat{H}_i y_i^2 + \hat{r}_i y_i\right),$$

where

$$\begin{aligned}\hat{H}_i &= H_{ii} + \sum_{k \in N(i)} \alpha_{ki}, \\ \hat{r}_i &= r_i + \sum_{k \in N(i)} \beta_{ki}.\end{aligned}$$

The estimated marginal mean is

$$\hat{y}_i = \frac{\hat{r}_i}{\hat{H}_i}.$$

The Gaussian Belief Propagation algorithm proceeds iteratively:

1. Initialize all messages α_{ij}, β_{ij} to zero.
2. For each edge (i,j) update messages using

$$\alpha_{ij} = -\frac{H_{ij}^2}{H_{ii} + \sum_{k \in N(i) \setminus j} \alpha_{ki}},$$

$$\beta_{ij} = -\frac{H_{ij}(r + \sum_{k \in N(i) \setminus j} \beta_{ki})}{H_{ii} + \sum_{k \in N(i) \setminus j} \alpha_{ki}}.$$

3. Repeat until convergence.
4. Compute marginal means via

$$\hat{y}_i = \frac{\hat{r}_i}{\hat{H}_i}.$$

These update equations show that:

- GaBP performs purely local computations.
- Each update depends only on neighboring nodes.
- No global matrix inversion is required.

4.4 Convergence Properties of Gaussian Belief Propagation

Although Gaussian Belief Propagation (GaBP) provides an elegant distributed method for solving linear systems, its convergence is not guaranteed for arbitrary matrices. In this section, we discuss sufficient conditions under which GaBP converges and produces the exact solution of the linear system

$$Hy = r,$$

where H is symmetric positive definite.

If the underlying graphical model is a tree, Gaussian Belief Propagation converges in a finite number of iterations and yields the exact marginal means. This follows from the general theory of Belief Propagation: on acyclic graphs, message passing performs exact inference. However, most linear systems of interest correspond to graphs containing cycles. In such cases, GaBP becomes an instance of loopy belief propagation, and convergence is not automatically guaranteed.

Several sufficient conditions for convergence have been established in the literature. A commonly cited sufficient condition is strict diagonal dominance of the matrix H :

$$|H_{ii}| > \sum_{j \neq i} |H_{ij}|, \forall i.$$

If this condition holds, GaBP converges and produces the exact solution. More generally, convergence can be characterized using spectral properties of the matrix. Define the matrix splitting

$$H = D - R,$$

where D is the diagonal of H and $R = D - H$.

GaBP converges if the spectral radius satisfies $\rho(D^{-1}R) < 1$.

This condition is analogous to convergence criteria for classical iterative methods such as Jacobi or Gauss–Seidel iterations.

5. GaBP-Based Interior-Point Method

5.1 Primal-Dual Interior-Point Formulation

In this section, we extend the discussion to the primal-dual interior-point method, which is the standard and most efficient practical implementation of interior-point algorithms.

Recall the primal linear programming problem:

$$\begin{aligned} \min_x \quad & c^T x \\ \text{subject to} \quad & Ax = b, x \geq 0. \end{aligned}$$

The corresponding dual problem is

$$\begin{aligned} \max_{y,s} \quad & b^T y \\ \text{subject to} \quad & A^T y + s = c, s \geq 0. \end{aligned}$$

where:

- y are dual variables associated with equality constraints,
- s are slack variables.

Optimality of primal and dual solutions is characterized by the KKT conditions:

$$\begin{aligned} Ax &= b, \\ A^T y + s &= c, \\ x_i s_i &= 0, \forall i, \\ x &\geq 0, s \geq 0. \end{aligned}$$

The complementarity condition $x_i s_i = 0$ ensures that at optimality at least one of each pair (x_i, s_i) is zero.

Interior-point methods replace the complementarity condition with the perturbed relation

$$x_i s_i = \mu,$$

where $\mu > 0$ is the barrier parameter.

In vector form:

$$Xs = \mu \mathbf{1}.$$

Thus, the primal-dual system becomes:

$$\begin{aligned} Ax &= b, \\ A^T y + s &= c, \\ Xs &= \mu \mathbf{1}. \end{aligned}$$

This nonlinear system defines the primal-dual central path.

Compared to the purely primal barrier method, the primal-dual formulation:

- exhibits superior numerical stability,
- converges faster in practice,
- directly enforces both primal and dual feasibility.

To solve this nonlinear system, Newton's method is applied to the full KKT system.

5.2 Newton System in Primal-Dual Form

Based on the primal-dual system defined in previous section, we define the residuals:

$$\begin{aligned} r_p &= Ax - b, \\ r_d &= A^T y + s - c, \\ r_c &= Xs - \mu \mathbf{1}. \end{aligned}$$

At each iteration, Newton's method computes corrections $(\Delta x, \Delta y, \Delta s)$ by linearizing the residual equations.

Linearizing each equation gives:

Primal feasibility:

$$A\Delta x = -r_p.$$

Dual feasibility:

$$A^T \Delta y + \Delta s = -r_d.$$

Complementarity condition:

Since $Xs = \mu \mathbf{1}$, its differential is

$$S\Delta x + X\Delta s = -r_c,$$

Where $S = \text{diag}(s)$.

Collecting all equations, we obtain the linear system

$$\begin{bmatrix} 0 & A^T & I \\ A & 0 & 0 \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = - \begin{bmatrix} r_d \\ r_p \\ r_c \end{bmatrix}.$$

This is the full Newton system for the primal-dual method. To obtain a system suitable for numerical solution, we eliminate Δs .

From the dual feasibility equation:

$$\Delta s = -r_d - A^T \Delta y.$$

Substituting into the complementarity equation:

$$S\Delta x + X(-r_d - A^T \Delta y) = -r_c.$$

Rearranging:

$$S\Delta x - XA^T \Delta y = -r_c + Xr_d.$$

Solving for Δx :

$$\Delta x = S^{-1}(-r_c + Xr_d + XA^T \Delta y).$$

Substitute this expression for Δx into the primal feasibility equation

$$A\Delta x = -r_p.$$

We obtain

$$AS^{-1}XA^T\Delta y = -r_p - AS^{-1}(-r_c + Xr_d).$$

Thus, the reduced Newton system becomes

$$H\Delta y = \tilde{r},$$

Where $H = AS^{-1}XA^T$ and $\tilde{r}$ is the modified residual vector. The matrix H is symmetric positive definite. Observe that:

- X and S are diagonal matrices.
- $S^{-1}X$ is diagonal and positive definite.
- Therefore, H has the same structural sparsity pattern as AA^T .

This structure makes the system suitable for Gaussian graphical modeling. Indeed, the matrix H plays the role of the precision matrix in the Gaussian interpretation.

We have shown that the primal-dual Newton step reduces to solving a symmetric positive definite linear system of the form

$$H\Delta y = \tilde{r}.$$

This system has the same structure as the least-squares normal equations derived earlier. Therefore, it can be solved using Gaussian Belief Propagation.

5.3 GaBP Solver for the Newton Step

We associate the linear system

$$H\Delta y = \tilde{r}$$

with the Gaussian distribution

$$p(\Delta y) \propto \exp\left(-\frac{1}{2}\Delta y^T H \Delta y + \tilde{r}^T \Delta y\right).$$

In this representation:

- The precision matrix is H .
- The information vector is $\tilde{r}$.

Computing the MAP estimate of this Gaussian distribution yields

$$\Delta y^* = H^{-1} \tilde{r},$$

which is exactly the solution of the Newton system.

The sparsity pattern of H determines the structure of the Gaussian graphical model.

- Each dual variable Δy_i corresponds to a node.
- An edge exists between nodes i and j if $H_{ij} \neq 0$.

Since

$$H = AS^{-1}XA^T,$$

its sparsity pattern is governed by the structure of the constraint matrix A . If A is sparse, the resulting graphical model is also sparse.

To compute Δy , we apply the GaBP message updates derived in Chapter 4. Each node i maintains:

- local precision H_{ii} ,
- local information $\tilde{r}_i$.

For each edge (i, j) , messages are updated iteratively:

$$\alpha_{ij} = -\frac{H_{ij}^2}{H_{ii} + \sum_{k \in N(i) \setminus j} \alpha_{ki}},$$

$$\beta_{ij} = -\frac{H_{ij}(\tilde{r}_i + \sum_{k \in N(i) \setminus j} \beta_{ki})}{H_{ii} + \sum_{k \in N(i) \setminus j} \alpha_{ki}}.$$

After convergence, the marginal means are computed as

$$\Delta y_i = \frac{\tilde{r}_i + \sum_{k \in N(i)} \beta_{ki}}{H_{ii} + \sum_{k \in N(i)} \alpha_{ki}}.$$

If GaBP converges, this solution coincides with the exact Newton direction. Once Δy is computed, the remaining Newton directions are obtained via back-substitution. From Section 5.2:

$$\Delta x = S^{-1}(-r_c + Xr_d + XA^T \Delta y),$$

and

$$\Delta s = -r_d - A^T \Delta y.$$

Thus, only the reduced system in Δy requires solving a linear system; all other variables follow from explicit formulas.

Replacing a direct solver with GaBP introduces:

- Iterative message passing instead of matrix factorization.
- Local computations per edge.
- Potential parallelization across nodes.

The computational complexity per iteration depends on:

- the number of nonzero entries in H ,
- the number of GaBP iterations required for convergence.

When the matrix is sparse and well conditioned, GaBP provides an efficient distributed alternative to classical Newton solvers.

5.4 Complete BP-Based LP Algorithm

Given the linear program

$$\begin{aligned} \min_x \quad & c^T x \\ \text{subject to} \quad & Ax = b, x \geq 0, \end{aligned}$$

we solve the perturbed KKT system

$$\begin{aligned}
Ax &= b, \\
A^T y + s &= c, \\
Xs &= \mu \mathbf{1}.
\end{aligned}$$

At each iteration, we compute a Newton direction using GaBP.

BP based Interior-Point Algorithm

Input: Matrix A , vectors b, c , initial strictly feasible (x^0, y^0, s^0) .

Parameters: Tolerance ϵ , barrier reduction factor $\sigma \in (0,1)$.

Step 0: Initialization

- Set $k = 0$.
- Choose initial strictly feasible $x^0 > 0, s^0 > 0$.
- Set initial barrier parameter

$$\mu^0 = \frac{(x^0)^T s^0}{n}.$$

Step 1: Compute Residuals

$$\begin{aligned}
r_p &= Ax^k - b, \\
r_d &= A^T y^k + s^k - c, \\
r_c &= X^k s^k - \mu^k \mathbf{1}.
\end{aligned}$$

If $\|r_p\| + \|r_d\| + \|r_c\| < \epsilon$, terminate.

Step 2: Form Reduced Newton System

Construct the matrix

$$H = AS^{-1}XA^T,$$

and the right-hand side

$$\tilde{r} = -r_p - AS^{-1}(-r_c + Xr_d).$$

Step 3: Solve Reduced System Using GaBP

Apply Gaussian Belief Propagation to solve

$$H\Delta y = \tilde{r}.$$

Obtain the Newton direction Δy .

Step 4: Recover Remaining Directions

$$\Delta x = S^{-1}(-r_c + Xr_d + XA^T\Delta y),$$

$$\Delta s = -r_d - A^T\Delta y.$$

Step 5: Line Search and Update

Choose step size $\alpha \in (0,1]$ to maintain positivity:

$$x^{k+1} = x^k + \alpha\Delta x,$$

$$y^{k+1} = y^k + \alpha\Delta y,$$

$$s^{k+1} = s^k + \alpha\Delta s.$$

Update barrier parameter

$$\mu^{k+1} = \sigma \frac{(x^{k+1})^T s^{k+1}}{n}.$$

Set $k = k + 1$ and repeat.

The only difference compared to classical implementations is Step 3:

- Instead of solving the Newton system using Cholesky factorization,
- We solve it via Gaussian Belief Propagation.

The algorithm alternates between:

1. Forming a Newton system.
2. Interpreting it as a Gaussian graphical model.
3. Performing probabilistic inference via message passing.
4. Updating primal-dual variables.

Thus, the linear algebraic core of interior-point optimization is replaced by distributed probabilistic inference.

This construction demonstrates that:

- Interior-point methods can be reinterpreted within graphical model theory.
- Newton steps correspond to Gaussian inference problems.
- Belief Propagation provides an alternative solver for LP problems.

5.5 Computational Considerations

The computational efficiency of the proposed method depends critically on the sparsity of the matrix

$$H = AS^{-1}XA^T.$$

Since X and S^{-1} are diagonal matrices, the sparsity pattern of H is determined entirely by the structure of the constraint matrix A .

If A is sparse:

- H is sparse,
- the corresponding Gaussian graphical model is sparse,
- each node has relatively few neighbors,
- GaBP message updates are computationally inexpensive.

Therefore, the method is particularly suitable for large-scale sparse linear programs.

In classical primal-dual interior-point methods, solving the Newton system typically requires:

- Cholesky factorization of H ,
- computational complexity depending on fill-in and matrix ordering.

In contrast, GaBP:

- performs iterative local updates,

- avoids explicit matrix factorization,
- can be parallelized across nodes.

The per-iteration cost of GaBP depends on:

- the number of edges in the graph (nonzero off-diagonal entries of H),
- the number of message-passing iterations required for convergence.

If convergence is achieved in a small number of iterations, GaBP may provide a competitive alternative to direct solvers.

The matrix H is symmetric positive definite by construction. However, as the barrier parameter decreases, the conditioning of the system may deteriorate.

This affects:

- the convergence rate of GaBP,
- numerical stability of the updates,
- sensitivity to round-off errors.

In practice, convergence may be improved by:

- damping message updates,
- appropriate scaling of the matrix,
- careful initialization of messages.

These techniques are commonly used in loopy belief propagation to enhance stability.

Compared to direct linear solvers, the GaBP-based approach has the following characteristics.

Advantages:

- avoids explicit matrix inversion,
- naturally distributed and parallelizable,
- leverages sparsity structure directly.

Potential limitations:

- convergence is not guaranteed for arbitrary matrices,
- may require more iterations than direct factorization,
- performance depends on conditioning of the system.

Thus, GaBP should be viewed as an alternative computational strategy rather than a universal replacement for classical methods.

6. Convergence Analysis

6.1 Convergence of Gaussian Belief Propagation

Gaussian Belief Propagation (GaBP) is an iterative algorithm for solving symmetric positive definite linear systems through message passing on a graphical model. Although GaBP yields the exact solution when it converges, convergence is not guaranteed for arbitrary matrices so in this section, we review sufficient conditions under which GaBP converges.

GaBP can be viewed as a nonlinear iterative mapping on the space of message parameters. Let

$$Ay = b$$

be a symmetric positive definite linear system. GaBP generates a sequence of message updates that, if convergent, produce marginal means satisfying

$$y^* = A^{-1}b.$$

Thus, the convergence of GaBP corresponds to convergence of its message parameters to a fixed point. A classical sufficient condition for GaBP convergence is strict diagonal dominance of the precision matrix:

$$|A_{ii}| > \sum_{j \neq i} |A_{ij}|, \forall i.$$

Under this condition, GaBP converges and yields the exact solution. More generally, convergence can be characterized using matrix splitting techniques. Consider the decomposition

$$A = D - R,$$

where D is the diagonal of A , and $R = D - A$. GaBP converges if the spectral radius satisfies

$$\rho(D^{-1}R) < 1,$$

where $\rho(\cdot)$ denotes the spectral radius. This condition resembles convergence criteria for classical iterative solvers such as Jacobi iteration.

It has been shown that GaBP is closely related to classical iterative methods for solving linear systems. In particular:

- The fixed point of GaBP corresponds to the solution of $Ay = b$.
- The convergence condition resembles that of Jacobi iteration.
- Under certain conditions, GaBP can be interpreted as a reparameterized linear iterative scheme.

However, unlike classical methods, GaBP derives its updates from probabilistic marginalization rather than purely algebraic matrix splitting.

In the BP-based interior-point framework, the Newton system matrix $H = AS^{-1}XA^T$ is symmetric positive definite. Since interior-point iterates remain strictly positive, both X and S are positive diagonal matrices.

Therefore, the reduced Newton system inherits positive definiteness, which is a necessary condition for GaBP applicability. However, positive definiteness alone does not guarantee convergence. The conditioning of the matrix plays an important role, especially as the barrier parameter decreases.

6.2 Exactness of the Fixed Point

Recall that the linear system $Ay = b$ is equivalent to minimizing the quadratic function

$$\phi(y) = \frac{1}{2}y^T Ay - b^T y.$$

The first-order optimality condition is

$$\nabla\phi(y) = Ay - b = 0.$$

Thus, the unique minimizer satisfies $y^* = A^{-1}b$. Since A is symmetric positive definite, this minimizer is unique.

In Gaussian Belief Propagation, each node i maintains a belief of the form

$$b_i(y_i) \propto \exp\left(-\frac{1}{2}\hat{A}_i y_i^2 + \hat{b}_i y_i\right).$$

At convergence, the marginal mean estimate is

$$\hat{y}_i = \frac{\hat{b}_i}{\hat{A}_i}.$$

If all messages stabilize, the belief parameters satisfy consistency conditions across the graph.

These consistency conditions imply that the set of marginal means $\hat{y}$ satisfies the global optimality condition $A\hat{y} = b$.

The key theoretical result underlying GaBP is:

If GaBP converges, then the resulting marginal means are equal to the exact solution of the linear system.

The intuition behind this result is that:

- The Gaussian graphical model represents a strictly convex quadratic energy.
- The MAP estimate is unique.
- Any consistent set of local beliefs that satisfies the fixed-point message equations must correspond to the global minimizer.

Therefore, convergence implies exactness.

In the BP-based interior-point framework, GaBP is used to solve the reduced Newton system

$$H\Delta y = \tilde{r}.$$

If GaBP converges at each iteration, then $\Delta y = H^{-1}\tilde{r}$, which is exactly the Newton direction.

Thus, the BP-based solver produces the same update directions as a classical interior-point method. The only difference lies in the computational mechanism used to obtain the solution.

6.3 Convergence within the Primal-Dual Framework

It is well known that classical primal-dual interior-point methods converge under standard assumptions:

- The linear program is strictly feasible.
- The Newton system is solved exactly at each iteration.
- Step sizes preserve positivity of x and s .

Under these conditions, the iterates remain on the central path and converge to the optimal solution.

Thus, exact computation of the Newton direction guarantees global convergence of the optimization algorithm.

In the proposed framework, the Newton system $H\Delta y = \tilde{r}$ is solved using GaBP instead of a direct solver.

Two cases may occur:

Case 1: GaBP Converges

If GaBP converges at each iteration, then by Section 6.2,

$$\Delta y = H^{-1}\tilde{r}.$$

In this case, the Newton direction is computed exactly, and the overall algorithm behaves identically to a classical primal-dual interior-point method. Therefore, convergence to the optimal LP solution is preserved.

Case 2: GaBP Does Not Fully Converge

If GaBP does not converge or is terminated early, the computed direction is only approximate.

In this situation:

- The update direction may deviate from the exact Newton direction.
- The algorithm behaves similarly to an inexact Newton method.

Interior-point methods are known to tolerate inexact solutions of the Newton system, provided that the residual error is sufficiently controlled. Thus, even when GaBP produces an approximate solution, convergence of the outer optimization loop may still be maintained, although possibly at a slower rate.

As the barrier parameter μ decreases, the matrices X and S become increasingly ill-conditioned near optimality. Consequently, the reduced Newton matrix $H = AS^{-1}XA^T$ may also become ill-conditioned.

Poor conditioning can negatively affect: GaBP convergence speed, numerical stability, and accuracy of the computed direction.

This behavior is not unique to the BP-based method; classical solvers also experience conditioning challenges near optimality.

The convergence of the complete BP-based interior-point method depends on two interacting processes:

1. Convergence of GaBP at each Newton step.
2. Global convergence of the primal-dual interior-point iterations.

When GaBP converges reliably, the overall method inherits the convergence guarantees of classical interior-point algorithms. When GaBP is used as an approximate solver, the method behaves as an inexact interior-point scheme.

We conclude that:

- Gaussian Belief Propagation provides an exact solver for the Newton system when convergence occurs.
- The BP-based interior-point method retains theoretical convergence properties of classical methods under exact Newton directions.
- In practice, convergence behavior depends on spectral properties and conditioning of the reduced Newton matrix.

7. Numerical Example

7.1 Definition of an Illustrative Linear Programming Problem

In this chapter, we present a simple linear programming example illustrating the complete BP-based interior-point framework developed in this thesis. The purpose of the example is to demonstrate the full transformation chain

$$\text{LP} \rightarrow \text{Primal-Dual Newton System} \rightarrow \text{Gaussian Model} \rightarrow \text{GaBP}.$$

Consider the linear programming problem in standard form

$$\min_{x \in \mathbb{R}^3} c^T x$$

subject to

$$Ax = b, x \geq 0,$$

where

$$c = \begin{bmatrix} 1 \\ 1 \\ 3 \end{bmatrix}, A = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}, b = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

The constraints are therefore

$$x_1 + x_2 = 1, x_2 + x_3 = 1, x_i \geq 0.$$

A strictly feasible point is

$$x^{(0)} = \begin{bmatrix} 0.5 \\ 0.5 \\ 0.5 \end{bmatrix},$$

since

$$Ax^{(0)} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad x^{(0)} > 0.$$

Using the constraints,

$$x_1 = 1 - x_2, x_3 = 1 - x_2,$$

and therefore

$$c^T x = x_1 + x_2 + 3x_3 = (1 - x_2) + x_2 + 3(1 - x_2) = 4 - 3x_2.$$

Minimizing the objective is therefore equivalent to maximizing x_2 on the interval $0 \leq x_2 \leq 1$. The optimal solution is

$$x^* = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}.$$

The corresponding dual problem is

$$\max_{y,s} b^T y$$

subject to

$$A^T y + s = c, s \geq 0.$$

7.2 Construction of the Primal-Dual Newton System

We consider the initial primal-dual point

$$x^{(0)} = \begin{bmatrix} 0.5 \\ 0.5 \\ 0.5 \end{bmatrix}, \quad y^{(0)} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

From the dual feasibility condition

$$A^T y + s = c$$

we obtain

$$s^{(0)} = c - A^T y^{(0)} = \begin{bmatrix} 1 \\ 1 \\ 3 \end{bmatrix}.$$

Thus

$$x^{(0)} > 0, s^{(0)} > 0.$$

The barrier parameter is

$$\mu^{(0)} = \frac{(x^{(0)})^T s^{(0)}}{3} = \frac{2.5}{3} = \frac{5}{6}.$$

The residuals are

$$\begin{aligned} r_p &= Ax^{(0)} - b = 0, \\ r_d &= A^T y^{(0)} + s^{(0)} - c = 0. \end{aligned}$$

The complementarity residual is

$$r_c = X^{(0)} s^{(0)} - \mu^{(0)} \mathbf{1}.$$

Since

$$X^{(0)} = \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{bmatrix},$$

we obtain

$$X^{(0)} s^{(0)} = \begin{bmatrix} 0.5 \\ 0.5 \\ 1.5 \end{bmatrix}.$$

Hence

$$r_c = \begin{bmatrix} \frac{1}{3} \\ -\frac{1}{3} \\ \frac{2}{3} \end{bmatrix}.$$

The reduced Newton matrix is

$$H = AS^{-1}XA^T.$$

Since

$$S^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{1}{3} \end{bmatrix},$$

we obtain

$$S^{-1}X = \begin{bmatrix} \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{6} \end{bmatrix}.$$

Thus

$$H = \begin{bmatrix} 1 & \frac{1}{2} \\ \frac{1}{2} & \frac{2}{3} \end{bmatrix}.$$

The reduced right-hand side is

$$\tilde{r} = -AS^{-1}(-r_c).$$

Since

$$-r_c = \begin{bmatrix} \frac{1}{3} \\ \frac{1}{3} \\ \frac{2}{3} \\ -\frac{1}{3} \end{bmatrix},$$

we obtain

$$\tilde{r} = \begin{bmatrix} \frac{2}{3} \\ -\frac{1}{9} \\ \frac{1}{3} \\ -\frac{2}{9} \end{bmatrix}.$$

The reduced Newton system therefore becomes

$$\begin{bmatrix} 1 & \frac{1}{2} \\ \frac{1}{2} & \frac{2}{3} \end{bmatrix} \Delta y = \begin{bmatrix} -\frac{2}{3} \\ -\frac{1}{9} \end{bmatrix}.$$

The solution is

$$\Delta y = \begin{bmatrix} -\frac{14}{15} \\ \frac{8}{15} \end{bmatrix}.$$

7.3 Gaussian Interpretation and GaBP Computation

The reduced Newton system

$$H\Delta y = \tilde{r}$$

can be interpreted as the MAP estimation problem for the Gaussian distribution

$$p(\Delta y) \propto \exp\left(-\frac{1}{2}\Delta y^T H \Delta y + \tilde{r}^T \Delta y\right).$$

The precision matrix is

$$H = \begin{bmatrix} 1 & \frac{1}{2} \\ \frac{1}{2} & \frac{2}{3} \end{bmatrix},$$

and the information vector is

$$\tilde{r} = \begin{bmatrix} \frac{2}{3} \\ -\frac{1}{9} \end{bmatrix}.$$

Since the off-diagonal entries of H are nonzero, the corresponding Gaussian graphical model contains two connected nodes representing the variables Δy_1 and Δy_2 .

Applying the GaBP message update equations derived in Chapter 4 yields the messages

$$\begin{aligned} \alpha_{12} &= -\frac{1}{4}, \beta_{12} = \frac{1}{3}, \\ \alpha_{21} &= -\frac{3}{8}, \beta_{21} = \frac{1}{12}. \end{aligned}$$

The belief parameters are therefore

$$\begin{aligned} \hat{A}_1 &= \frac{5}{8}, \hat{b}_1 = -\frac{7}{12}, \\ \hat{A}_2 &= \frac{5}{12}, \hat{b}_2 = \frac{2}{9}. \end{aligned}$$

The marginal mean estimates become

$$\Delta y_1 = -\frac{14}{15}, \Delta y_2 = \frac{8}{15}.$$

These values coincide exactly with the solution of the linear system.

7.4 Recovery of the Newton Directions

The remaining Newton directions are obtained by back-substitution.

First,

$$\Delta s = -A^T \Delta y.$$

Thus

$$\Delta s = \begin{bmatrix} \frac{14}{15} \\ \frac{2}{5} \\ \frac{8}{15} \end{bmatrix}.$$

Next, $\Delta x = S^{-1}(-r_c + XA^T \Delta y)$.

This yields

$$\Delta x = \begin{bmatrix} -\frac{2}{15} \\ \frac{2}{15} \\ -\frac{2}{15} \end{bmatrix}.$$

Taking a full step $\alpha = 1$ gives

$$x^{(1)} = \begin{bmatrix} \frac{11}{30} \\ \frac{19}{30} \\ \frac{11}{30} \end{bmatrix},$$

$$y^{(1)} = \begin{bmatrix} -\frac{14}{15} \\ \frac{8}{15} \end{bmatrix},$$

$$s^{(1)} = \begin{bmatrix} \frac{29}{15} \\ \frac{7}{5} \\ \frac{37}{15} \end{bmatrix}.$$

The new primal iterate satisfies

$$x_2^{(1)} > x_2^{(0)},$$

indicating progress toward the optimal solution.

8. Conclusion

In this thesis, we studied a probabilistic reinterpretation of the interior-point method for solving linear programming problems, originally proposed by Bickson et al. [1]. The main objective was to analyze how the Newton system arising in interior-point optimization can be reformulated as a probabilistic inference problem in a Gaussian graphical model and solved using Gaussian Belief Propagation (GaBP).

We began by deriving the classical barrier and primal-dual formulations of linear programming and showed that the Newton step reduces to solving a symmetric positive definite linear system. This system was then expressed as a least-squares problem and subsequently reinterpreted as a quadratic optimization problem. By recognizing that quadratic objectives correspond to the negative logarithm of Gaussian distributions in information form, we showed how the equivalence between solving linear systems and computing MAP estimates in Gaussian models arises naturally from the probabilistic formulation.

Building on this equivalence, we constructed the associated Gaussian graphical model and derived the Gaussian Belief Propagation message update equations. We showed that, when GaBP converges, its fixed point coincides exactly with the solution of the Newton system. Consequently, the Newton direction required by the primal-dual interior-point method can be obtained through distributed probabilistic inference instead of direct matrix factorization.

The proposed framework therefore provides:

- A conceptual bridge between convex optimization and probabilistic graphical models,
- A distributed and message-passing-based alternative to classical linear system solvers,
- A unified interpretation of interior-point methods within the language of MAP inference.

The convergence analysis demonstrated that GaBP produces the exact solution whenever convergence occurs, and that the overall optimization method retains the convergence properties of classical interior-point algorithms under exact or sufficiently accurate Newton directions.

A simple numerical example illustrated the full transformation chain:

LP $\rightarrow$ Primal-Dual Newton System $\rightarrow$ Gaussian Graphical Model $\rightarrow$ Belief Propagation System

Although the example was elementary, it clearly demonstrated the structural equivalence underlying the proposed method.

Several directions for future research arise from this work:

- A deeper theoretical study of GaBP convergence within interior-point iterations,
- Development of damping or preconditioning strategies to improve robustness,
- Empirical comparison with classical solvers on large sparse LP problems,
- Extension of the framework to quadratic programming or more general convex optimization problems.

The framework studied in this thesis is based on the approach originally proposed by Bickson et al. (2008), which established a connection between interior-point methods and Gaussian Belief Propagation. The contribution of this thesis lies in providing a detailed theoretical analysis, systematic derivation, and illustrative implementation of this framework. The obtained results highlight the close relationship between convex optimization, numerical linear algebra, and probabilistic graphical models.

References

- [1] D. Bickson, Y. Tock, O. Shental, and D. Dolev, “Polynomial Linear Programming with Gaussian Belief Propagation,” *Proceedings of the 46th Annual Allerton Conference on Communication, Control, and Computing*, 2008.
- [2] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.
- [3] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*, MIT Press, 2009.

Biography

Ivan Hornjak was born on 11.08.1993 in Novi Sad. He attended elementary school Ivo Lola Ribar in Novi Sad and later graduated from secondary school Svetozar Miletić.

In 2012, he enrolled at the Faculty of Sciences, University of Novi Sad, where he began his studies in Applied mathematics – Financial mathematics. After completing his undergraduate studies in 2017, he began his Master's studies in Applied Mathematics – Data Science at the same faculty.

Alongside his academic studies, he gained professional experience in the IT industry. He currently works in the field of data science and data analytics, with a particular focus on applying mathematical and statistical methods to real-world problems.