

# Cheapest Super-Replication of Vanilla Options

## *A Linear-Programming Approach to Super Replication and Arbitrage Detection*

*This project is slightly more ambitious than the two other proposals, and aimed at students already familiar with the C++ programming language. The added difficulty is that it requires students to make use of a third-party library.*

## 1 Introduction

Just like their underlyings, listed options are traded on exchanges with an order book mechanism, so that, in order to guarantee a transaction, buyers must cross the bid-ask spreads of the considered instruments.

In this project, we consider the case of a trader interested in buying a European option with a given maturity  $T$  with a piecewise linear payoff.

A piecewise-linear payoff may be written as a linear combination of Call and Put payoffs. A buyer willing to purchase the corresponding linear combination must trade Calls and Puts at the Ask price when the corresponding weight is positive, and the Bid price when the corresponding weight is negative. This can be prohibitive, especially when certain strikes are illiquid which results in large bid-ask spreads.

The goal of this project is to produce a routine that outputs the weights for a linear combination of calls and puts that super-replicates the target payoff while minimizing the cost.

## 2 C++ Computing Project

Produce a C++ routine outputting weights for the super replication of a piecewise linear function of compact support with Call options.

*This step should require no optimization whatsoever.*

Formulate the problem of the *minimal super replication* of a piecewise linear profile with combination of calls and puts as a linear programming problem.

Using a third-party linear programming solver, or with your own implementation of the simplex method, provide a routine computing the cheapest super-replication price for any given piecewise linear profile.

What would it mean if the cheapest replication of the null payoff had a negative price?