电 子 科 技 大 学

实 验 报 告

学生姓名：　　　学　　号：　　　指导教师：

实验地点：　　　　　　实验时间：

一、实验室名称： 经济管理专业实验室

二、实验项目名称：黄金联结债券产品设计

三、实验学时： 4 学时

四、实验原理：

黄金联结债券(Gold-linked Notes, GLNs)一般是由规模较大的黄金生产企业按照规定的条件和程序发行，对投资者提供本金保护(Principal Protection）、一定收益率(Yield)，同时产品收益/风险与金价挂钩（Exposure to Gold Price Fluctuations)的新型投资产品。

黄金联结债券的实质是一种复合型结构(Hybrid)，其主要体现在以下方面：1）证券与商品的混合。其既是一个债券，同时又是一个和黄金价格相关的商品；2）普通债券和奇异期权(Exotic Options）的混合。其在债券里往往会嵌入(Embedded)一些期权，甚至是奇异期权。如：宏源证券发行的产品中，嵌入了奇异期权中的障碍期权(Barrier Options)。

对于Hybrid bond with embedded barrier option and cap的定价问题，可采用单因素模型，即标的黄金价格一个随机变量，使用Monte-Carlo Simulation对其进行定价研究。

首先，对标的黄金价格进行参数校准(Calibration)。

可通过历史黄金价格的走势分析，根据历史趋势的分析，通常黄金价格走势服从几何布朗运动(Geometric Brownian Motion)。

$$u_i = \ln\left(G_i / G_{i-1}\right), i = 1, 2$$

$$E\left(u\right) = \frac{1}{n}\sum_{i=1}^{n} u_i$$

$$\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left[u_i - E\left(u\right)\right]^2}$$

假设黄金价格服从如下随机过程：

$$dG = \mu Gdt + \sigma Gdz$$

其中，G为黄金价格，dz为标准维纳过程增量，μ为黄金价格漂移项，σ为黄金价格波动项。

其次，通过风险中性技术建模定价。

$$B\left(G\left(t\right), t\right) = e^{-r_f\left(T-t\right)} E_Q\left\{Payoff_T\right\}$$

其中，$B(G(t),t)$是债券价格，r_f是无风险折现率，E_Q表示在风险中性下的未来期望收益。$Payoff_T$表示未来债券在到期T时刻的收益。

然后设计$Payoff_T$：

第一年的收益：

$$Payoff_{t+1} = \left(1 + 3.5\%\right) \times F \qquad G_{t+1} < \left(1 + 3.5\%\right) \times G_t$$

其中，F为债券面值，G_t为债券发行时黄金基准价格，G_{t+1}为债券发行后1年时的黄金基准价格。

第二年的收益：

$$Payoff_{t+2} = \left(1 + 3.5\%\right) \times F \qquad G_{t+2} < \left(1 + 3.5\%\right) \times G_t$$

第三年的收益分三种情况：

$$Payoff_T = \left(1 + 3.5\%\right) \times F \qquad G_T < \left(1 + 3.5\%\right) \times G_t$$

$$Payoff_T = \left(1 + 3.5\%\right) \times F + 0.3 \times \left(G_T - \left(1 + 3.5\%\right) \times G_t\right)$$
$$\left(1 + 3.5\%\right) \times G_t < G_T < \left(1 + 60\%\right) \times G_t$$

$$Payoff_T = \left(1 + 3.5\%\right) \times F + 0.3 \times \left(\left(1 + 60\%\right) \times G_t - \left(1 + 3.5\%\right) \times G_t\right)$$
$$G_T > \left(1 + 60\%\right) \times G_t$$

第一种情况：当到期黄金基准价格低于发行黄金基准价格的3.5%时，障碍期权条款不触发，投资者获得3.5%的固定利率收益和本金；

第二种情况：当到期黄金基准价格高于发行黄金基准价格的3.5%时，触发障碍期权条款，投资者将与发行者按30%与70%的比例分享金价超过3.5%部分的收益。

第三种情况：当到期黄金基准价格高于发行黄金基准价格的60%时，投资者将与发行者按30%与70%的比例分享金价在3.5%到60%部分的收益。

最后，蒙特卡罗定价技术

Monte-Carlo 模拟在处理复杂的路径依赖问题(Path Dependent)上的优势使得很多奇异衍生产品(Exotic Derivatives)的定价成为可能。其基本思想是从初始时刻的标的资产价格开始,根据假定的随机路径（Random Path）模拟出标的资产的到期价值,计算出每个到期价格下证券的收益,求出其均值,再以无风险利率贴现,完成证券的定价。

具体过程如下：

（1）产生随机黄金价格路径(Paths)；

（2）根据利率条款、嵌入障碍期权条款、封顶条款，计算出相应的 Payoff；

（3）重复（1）-（2）步骤，得出足够多路径(如：10，000 条路径)所对应的黄金联结债券价格；

（4）计算所有路径对应黄金联结债券价格的算术平均值，并对其进行风险中性贴现。Monte-Carlo 模拟的精度是一个需要考虑的问题。

提高 Monte-Carlo 模拟的精度的方法：

1）通过增加蒙特卡洛模拟的路径数；

2）对偶变量技术（Antithetic Variable Technique）。前者要消耗大量的计算时间，我们采用对偶变量技术。在对偶变量技术中，一次模拟运算包括计算衍生产品的两个值，第一个值 F_1 是用通常的方法得到；第二个值 F_2 是通过改变所有标准正态分布样本的符号计算出的。假设通过模拟得到函数 $F = f$（x），模拟的路径数为 $2n$ 。

假设通过模拟得到函数 $F = f(x)$，模拟的路径数为 $2n$，则有：

$$\hat{F} = \frac{1}{2n}\sum_{i=1}^{2n} f\left(x_i\right)$$

同时也可以用 n 个变量及其对应的对偶变量 $-n$ 来得到：

$$\hat{F} = \frac{1}{n}\sum_{i=1}^{n} \frac{f\left(x_i\right) + f'\left(x_i\right)}{2}$$

五、实验目的：

掌握结构化衍生证券的基本原理、设计及定价方法。

六、实验内容：

假设债券面值为 F＝10000 元，当前黄金基准价格为 Gp=200 元/克，黄金价格波动率 σ=0.2315，无风险利率为 0.03，债券息票利率为 0.035，一年付息一次。债券收益同上文。

对该债券定价。

七、实验所用软件平台：Matlab 软件或其他软件

八、实验步骤：

- 熟悉算法
- 编写程序
- 调试
- 给出结果并分析

九、实验数据及结果分析（可另附页）：

根据实验原理，编写了 Python 程序进行蒙特卡洛模拟，采用了对偶变量的优化方式，得到了以下数据

```python
import math
import matplotlib.pyplot as plt
import scipy.special
from scipy import linalg
import numpy as np


def CalculateGoldPrice(OldPrice, r=0.03, dt=1/252, sigma=0.2315, ran=0):
    return OldPrice + r*OldPrice * \
        dt+sigma*OldPrice*ran*math.sqrt(dt)


def MonteCarlo(N_path=1000, Gp=200, sigma=0.2315, r=0.03, nr=0.035, F=10000, dt=1/252):
    plt.rc("font", family='YouYuan')  # 显示中文
    FT = np.full(N_path*2+1, F)  # 初始证券价格
    gpath = np.arange(1, N_path*2+1, 1).tolist()
    # print(FT)
    for num in range(1, N_path+1):  # 路径个数

        np.random.seed(num*2+1)  # 设置随机数种子

```

```python
        gold_price = np.zeros(252*3+1)
        gold_price_2 = np.zeros(252*3+1)
        gold_price[1] = Gp  # 黄金基准价格
        gold_price_2[1] = Gp
        day = np.arange(1, 252*3+1, 1).tolist()
        payoff = nr*F*(math.exp(-r) + math.exp(-r*2) +
                       math.exp(-r*3)) + F * math.exp(-r*3)  # 三年利息的现值 + 本金的现值
        for k in range(1, 252*3):
            ran = np.random.randn(1)
            ran_2 = -ran  # 获得对偶因子
            # 离散形式模拟金价
            gold_price[k+1] = CalculateGoldPrice(gold_price[k], ran=ran)
            gold_price_2[k+1] = CalculateGoldPrice(gold_price_2[k], ran=ran_2)
            if(k == 252*3-1):
                limit_1 = (1+nr)*gold_price[1]
                limit_2 = (1+0.6)*gold_price[1]
                if(gold_price[k+1] <= limit_1):
                    FT[num] = payoff + limit_1*math.exp(-r*3)
                if(limit_1 < gold_price[k+1] and gold_price[k+1] < limit_2):
                    FT[num] = payoff + limit_1*math.exp(-r*3) + 0.3 * \
                        (gold_price[k+1]-limit_1)*math.exp(-r*3)
                if(limit_2 <= gold_price[k+1]):
                    FT[num] = payoff + limit_1 * \
                        math.exp(-r*3) + 0.3*(limit_2 - limit_1)*math.exp(-r*3)
                # 计算路径 2
                if(gold_price_2[k+1] <= limit_1):
                    FT[num+N_path] = payoff + limit_1*math.exp(-r*3)
                if(limit_1 < gold_price_2[k+1] and gold_price_2[k+1] < limit_2):
                    FT[num+N_path] = payoff + limit_1*math.exp(-r*3) + 0.3 * \
                        (gold_price_2[k+1]-limit_1)*math.exp(-r*3)
                if(limit_2 <= gold_price_2[k+1]):
                    FT[num+N_path] = payoff + limit_1*math.exp(-r*3) + 0.3 * \
                        (limit_2 - limit_1)*math.exp(-r*3)
        plt.plot(day, gold_price[1:])
        plt.plot(day, gold_price_2[1:])
    plt.title('黄金价格模拟路径')
    plt.grid(True)
    plt.xlabel('时间')
    plt.ylabel('黄金价格')
    plt.show()
    ###
    plt.title('债券价值分布')
    plt.grid(True)
```

```
    plt.xlabel('样本路径')
    plt.ylabel('债券价值')
    plt.plot(gpath, FT[1:], 'xk')
    plt.show()
    ###
    x = np.arange(10315, 10350, 0.5).tolist()
    plt.title('债券价值频率')
    plt.grid(True)
    plt.xlabel('债券价值区间')
    plt.ylabel('个数')
    plt.hist(FT, x)
    plt.show()
    return


if __name__ == '__main__':
    MonteCarlo(N_path=1000)
```

结果如下：

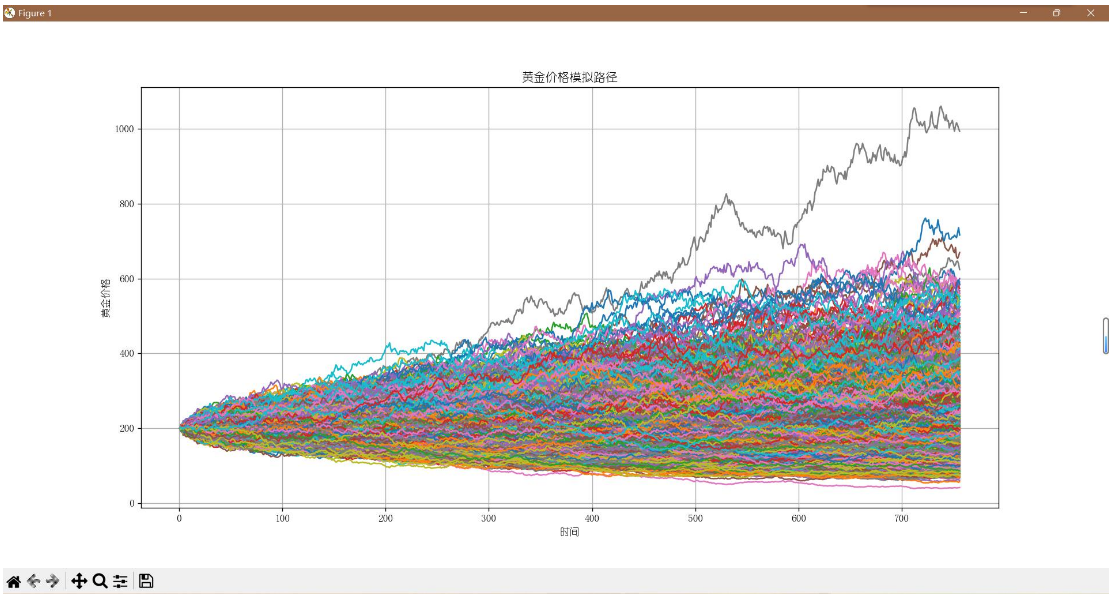

图 1　黄金价格模拟路径

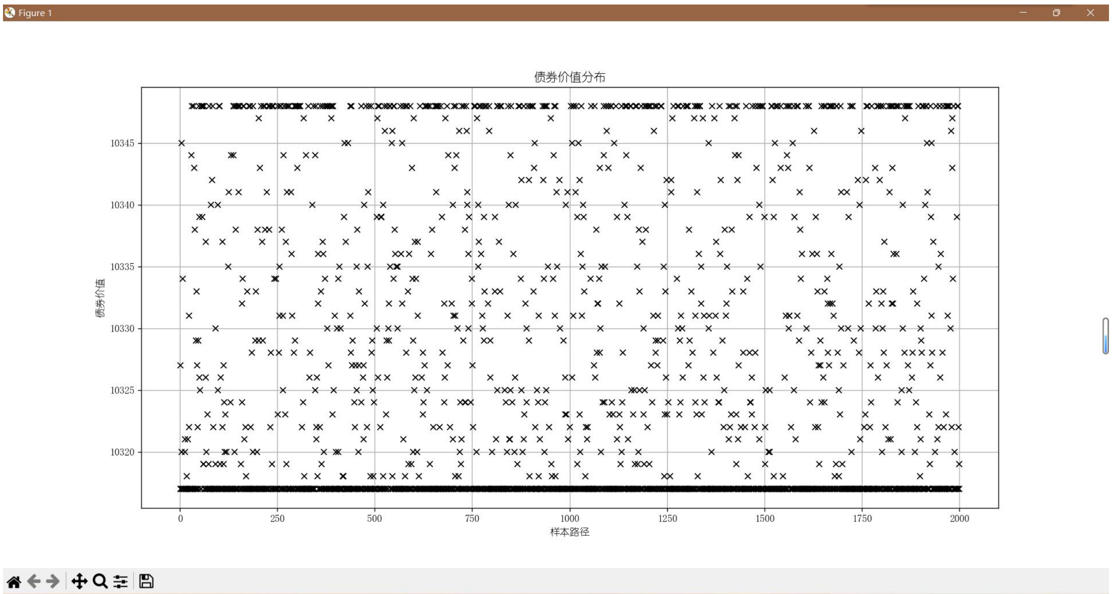

图 2　债券价值分布

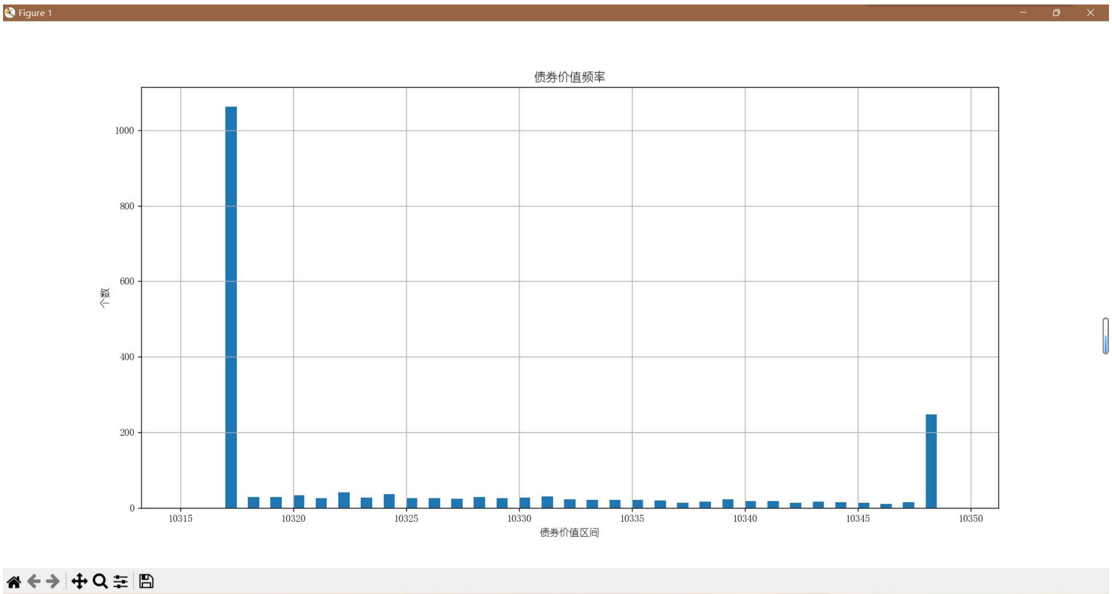

图 3　债券价值频率

十、实验结论：

定价结果主要分布在 10315 到 10350 之间。

十一、总结及心得体会：

通过本实验，我学会了如何通过程序，通过路径模拟的方式对结构化衍生证券进行定价，掌握了对服从几何布朗运动的价格模拟方法，巩固了风险中性定价原理。

在具体的编程中，我还学会了通过对偶变量的方式对程序进行优化，获得了

更高的模拟效率。学会了如何绘制折线图、散点图、直方图。

十二、对本实验过程及方法、手段的改进建议：

在本实验中，当我尝试路径数量大于 5000 时，程序效率变得很低，需要很久的时间运行。能否在定价时优化模拟模型，可以更快地算出结果。

报告评分：

指导教师签字：