Introduction

Option
- The right (not the obligation) to buy or sell an asset at a specific price on a certain date.
- Price = $E(\text{future payoff})$
- Can be priced by generating $n$ random payoffs and taking the sample mean.
- GAIL chooses $n$ automatically to achieve the desired error tolerance.

New MATLAB classes
- We have developed a class `optPrice` for pricing options using GAIL (Choi et al, 2015) (quasi-)Monte Carlo methods.
- `genOptPrice(...)` automatically computes the option price to the desired tolerance.
- Class definitions allow option types and stock path models to be added incrementally.
- I added three new option payoffs: digital, basket, and American.

Digital Option
Payoff if the final stock price exceeds the strike price.

Payoff
- Call: $P \times 1_{[K,\infty]}(S_T)$
- Put: $P \times 1_{[0,K]}(S_T)$

Digital Option Example

```matlab
% Payoff Parameters from optPayoff class
inp.payoffParam.optType = {'digitalcash'};
inp.payoffParam.putCallType = {'call'};
inp.payoffParam.strike = 12;
inp.payoffParam.digitalPay = 1;
% Asset Path Parameters from assetPath class
inp.assetParam.initPrice = 11;
inp.assetParam.interest = 0.01;
% Option Price Parameters
inp.priceParam.absTol = 0;
inp.priceParam.relTol = 0.002;
% Compute the Option Price
DigOption = optPrice(inp);
Price = genOptPrice(DigOption)
```

Basket Option
Based on several underlying assets.

Payoff
Determined by a weighted sum of the prices of the assets in the basket.

Call: $\max\left(\sum_{i=1}^{m} a_i \times S_T^{(i)} - K, 0\right)$

Put: $\max\left(K - \sum_{i=1}^{n} a_i \times S_T^{(i)}, 0\right)$

$m = \text{number of assets}$

Basket Option Example

```matlab
% Payoff Parameters:
inp.payoffParam.optType = {'basket'};
inp.payoffParam.basketWeight = [0.2 0.8];
% Asset Path Parameters:
inp.assetParam.initPrice = [11 15];
inp.assetParam.volatility = [0.5 0.6];
inp.assetParam.nAsset = 2;
inp.assetParam.corrMat = [1 0.5; 0.5 1];
% Compute the Option Price
BasketOption = optPrice(inp);
Price = genOptPrice(BasketOption)
```

American Option
Background
This type of option can be exercised at any time during its life. We apply the least squares method of Longstaff and Schwartz (2001) to price American put options in GAIL.

American Option Example

```matlab
% Payoff Parameters:
inp.payoffParam.optType = {'american'};
inp.payoffParam.putCallType = {'put'};
% Compute the Option Price
AmericanOption = optPrice(inp);
Price = genOptPrice(AmericanOption)
```

Discussion
- All of the three types of options are tested and results are within tolerance.
- Still working on American Option to fit for quasi-Monte Carlo Method.

References

tzhu8@hawk.iit.edu