Quantitative Finance

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/rquf20

A data-driven deep learning approach for options market making

Qianhui Lai, Xuefeng Gao & Lingfei Li

Quantitative Finance, 2023
Vol. 23, No. 5, 777–797, https://doi.org/10.1080/14697688.2023.2186257

A data-driven deep learning approach for options market making

QIANHUI LAI†, XUEFENG GAO‡ and LINGFEI LI‡*

†School of Economics, Qingdao University, Qingdao, People's Republic of China
‡Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, New Territories, Hong Kong SAR

(*Received 11 April 2022; accepted 24 February 2023; published online 16 March 2023*)

We develop a data-driven approach for options market making. Using stock options data from CBOE, we find that both buy and sell orders exhibit strong self-excitation but insignificant cross-excitation. We show that a Hawkes process with a time-varying baseline intensity and the power law kernel provides a good fit to the data of market order flows for stock options. To solve the optimal market making problem for a single option, we approximate the market making strategy at each decision time by a neural network and train them to optimize the expected utility of the market maker. We study feature selection for the neural networks and compare the out-of-sample performance of the optimal neural network strategy trained from data generated by the Hawkes process and the Poisson process. We find that using the more realistic Hawkes model improves the out-of-sample performance significantly. Furthermore, utilizing the Hawkes process intensity or the expected number of market order arrivals computed under the Hawkes model as an additional input feature can boost the performance. We also show how to solve the market making problem for option portfolios with Greeks and inventory constraints using neural network approximation.

Keywords: Market making; Algorithmic trading; Options; Deep learning; Hawkes process

1. Introduction

Market makers provide liquidity in securities markets by dynamically updating their bid and ask quotes to earn the bid-ask spread while mitigating the inventory risk. With the electronification of financial markets and the rise of high-frequency trading, the last two decades have witnessed a rapid growth in the study of the market making problem.

There is a large body of literature on market making strategies in equity and some over-the-counter markets. By combining the utility approach of Ho and Stoll (1981) and some empirical results from the econophysics literature, Avellaneda and Stoikov (2008) proposed a mathematical framework for high-frequency trading in an order-driven market and characterized the optimal market making strategy. Guéant *et al.* (2013) further introduced a bound on the inventory into the framework. Cartea and Jaimungal (2015, 2016) and Cartea *et al.* (2017, 2018a, 2018b) made a series of contributions by introducing new features like risk metrics for inventory risk, alpha and order book signals, aversion to model uncertainty, mutual excitation, etc. Guilbaud and Pham (2013, 2015) studied market making when the agent can set the limit orders at best bid (or ask) plus (or minus) a tick to obtain execution priority and in a pro rata microstructure, respectively. Abergel *et al.* (2020) proposed a market making model for a first-in-first-out limit order book. They modeled the intensities of limit orders, market orders and cancellation orders as functions of the state of the order book. Bergault and Guéant (2021) considered the multiasset market making problem and proposed a dimension reduction technique to solve it based on a factor model. Choi *et al.* (2021) introduced a synchronized factor in the market buy and sell orders. To obtain the optimal market making strategy, they solved a high-dimensinoal Hamilton-Jacobi-Bellman (HJB) equation by deep learning. Kumar (2021) studied market making in a limit order book model with event arrivals modeled by deep Hawkes processes. Several papers used reinforcement learning to solve the market making problem. Guéant and Manziuk (2019) developed a reinforcement learning algorithm to tackle the curse of dimensionality in market making for a large universe of corporate bonds in the Avellaneda-Stoikov framework. Zhao and Linetsky (2021) proposed to measure the adverse selection risk by the book exhaustion rate and used

*Corresponding author. Email: lfli@se.cuhk.edu.hk

it as a feature for reinforcement learning of the market making strategy. Gašperov and Kostanjčar (2021) constructed signals for market making using deep reinforcement learning with an adversary, while Gašperov and Kostanjčar (2022) solved the problem via deep reinforcement learning in a limit order book model based on multivariate Hawkes processes with exponential kernels. In addition, Gašperov *et al.* (2021) and Hambly *et al.* (2021) provided excellent reviews of reinforcement learning for market making.

Developing market making strategies for options contracts is undoubtedly an important problem in practice, but the literature on this topic is substantially smaller. To the best of our knowledge, the problem has only been addressed in Stoikov and Sağlam (2009), El Aoud and Abergel (2015), Baldacci *et al.* (2020) and Baldacci *et al.* (2021). Market making for options can be more complicated than for stocks, because it trades not only the options but also the underlying asset for the hedging purpose. Furthermore, one has to deal with multiple options of various strikes and maturities and consider constraints on the Greeks in addition to the inventory constraint.

In a pioneering work, Stoikov and Sağlam (2009) developed market making strategies for a single option under a mean-variance criterion in three settings: (1) a complete market with the stock assumed to be liquid and its price follows a geometric Brownian motion without drift; (2) an incomplete market that arises from the iliquidity of the stock which leads to market making in both the option and the stock; (3) an incomplete market with stochastic volatility and overnight jumps in the stock price. By assuming the execution intensity of limit orders linearly decays in the distance of the quote to the mid-price, they obtain analytical characterizations of the optimal bid and ask quotes in these settings. El Aoud and Abergel (2015) also considered a single option on a liquid stock, but they modeled the stock dynamics by a general stochastic volatility model and assumed that the execution intensity of limit orders decays as a power of the distance to the mid-price. They applied delta hedging and obtained analytical solutions and approximations for the optimal bid and ask quotes that maximize the expected utility of the terminal wealth in the risk-neutral and risk-averse case, respectively, for specific forms of the market impact function.

Baldacci *et al.* (2021) studied market making for multiple options on a single liquid stock in a general stochastic volatility framework. They applied delta hedging and assumed a general form for the execution intensity, which is a function of the distance to the mid-price. They considered two objectives for maximization: the expected utility of the terminal wealth or the expected wealth adjusted by inventory risk. This set-up leads to a high-dimensional control problem. To reduce the dimensionality, they approximated the option vega by a constant and obtained a low-dimensional HJB equation for the value function, which was solved by finite difference. Baldacci *et al.* (2020) considered a similar setting but dealt with options on several liquid stocks and the execution intensity is more general, which is a function of the stock price, volatility and distance to the mid-price. To tackle the high dimensionality, they approximated the value function as quadratic in the vector of inventories.

Our paper contributes to the options market making literature in the following ways.

(1) We provide an empirical study of the behavior of market order flows for options. The assumptions on the execution intensity in Stoikov and Sağlam (2009), El Aoud and Abergel (2015) and Baldacci *et al.* (2021) imply that market orders follow a Poisson process with a constant arrival rate, while in Baldacci *et al.* (2020), the arrival rate of options market orders could depend on state variables like stock price and volatility. In all these papers, the market orders do not show self-excitation. However, using a nonparametric approach we find significant self-excitation in the data for both market buy and sell orders but with substantially less cross-excitation between them. This motivates us to model market order arrivals using a Hawkes process (see e.g. Bacry *et al.* 2015 for an overview of Hawkes processes with financial applications) and we find that the power law kernel instead of the exponential kernel provides a good fit. This result highlights the non-Markov nature of the options order data, which the existing papers fail to account for. One can view our approach as data-driven, where the model is developed based on empirical facts as opposed to being specified a priori.

(2) The non-Markov intensity of market order arrivals which is fully path-dependent poses a challenge for solving the optimal control problem for market making. Confortola and Fuhrman (2013) represents the value functions of controlling non-Markov point processes by backward stochastic differential equations (BSDEs) driven by random measures. However, it is not clear how to solve such BSDEs and also deal with various contraints for market making. In this paper, we utilize deep learning to tackle the optimal control problem of options market making, which has two appealing features. First, it can accommodate general models for the stock price, the option market order arrivals and market impact, as long as we can simulate from them and calculate the option price in the stock model to generate data for training the neural networks. Second, it can easily handle the constraints on the inventory and Greeks by formulating them as penalties in the loss function for neural network training. To solve the problem by deep learning, we adopt the idea in Han and E (2016). At each decision time, we approximate the bid and ask quotes each by a feed-forward neural network (FNN) and select appropriate features as inputs to the neural networks. We then optimize over the parameters of the neural networks to find the optimal policy.

(3) We demonstrate the importance of using realistic models for learning the market making strategy. Whereas the existing papers conducted simulation studies to evaluate the performance of market making strategies, in this paper we perform an out-of-sample test based on real market data. Our finding suggests that the market making strategy learned based on a more realistic model can significantly outperform the one learned

based on a less realistic model. Furthermore, we show that using features from the Hawkes process that models the market order arrivals as inputs to the neural networks can boost the out-of-sample performance.

The rest of the paper is organized as follows. Section 2 provides an empirical analysis of market orders for stock options. Section 3 presents the market making problem for a single option and develops a deep learning algorithm to solve it. For simplicity, we do not consider constraints in this section. Section 4 evaluates the performance of several market making strategies on real market data. Section 5 solves the market making problem for multiple options with constraints on the inventory and Greeks by deep learning. The last section concludes the paper.

2. Empirical analysis of market orders for stock options

We analyze stock options in this paper. We acquired data of call options on two stocks, Apple (AAPL) and Bank of America (BAC) from Chicago Board Options Exchange (CBOE). Our data spans the whole year of 2019 for a total of 252 trading days. The APPL options are among the most liquid equity options traded on CBOE with average daily trading volume at 85 852 contracts in 2019. In comparison, the BAC options can be considered medium liquid, with average daily trading volume at 36,789 contracts in 2019, which is about half of the size of AAPL. Our research budget limits us from acquiring options data for more stocks, but the data for these two stocks is representative for obtaining meaningful results.

CBOE provides two datasets. The first one is for options trades which contains the following information for each trade: quote datetime, expiration, strike, option type, trade size and price, best bid and ask, implied volatility, delta and underlying bid and ask. The other one provides information at one minute frequency for all available options with variables listed above plus Greeks like gamma, theta, vega and rho. We will use the first dataset for analyzing market orders and the second one for inputs to the neural networks.

The CBOE data does not directly inform the trade direction, i.e. whether a trade is a buy or a sell. To infer it from the given data, we apply the method developed in Lee and Ready (1991). It is shown in Savickas and Wilson (2003) that the Lee-Ready method can correctly classify 80% of the trades for CBOE options data, which compares favorably with some alternative methods.

We calculate the number of market orders on each trading day based on the trade dataset. Figure 1 shows the average daily number of market orders for options with moneyness $K/S \in (0.8, 1.2]$ and time to maturity (TTM) up to 6 months. We can clearly observe that short-maturity options with moneyness close to at the money are more liquid than the others. As the empirical analysis requires sufficient data, in the following we will only analyze AAPL call options with moneyness in (0.99, 1.02] and TTM up to 10 days and BAC call options with moneyness in (0.98, 1.02] and TTM up to 5 days. We further partition them into a total of eight groups for analysis:

- Six groups of AAPL call options with $K/S \in (0.99, 1]$, TTM $\in (0, 5]$; $K/S \in (0.99, 1]$, TTM $\in [6, 10]$; $K/S \in (1, 1.01]$, TTM $\in (0, 5]$; $K/S \in (1, 1.01]$, TTM $\in [6, 10]$; $K/S \in (1.01, 1.02]$, TTM $\in (0, 5]$; $K/S \in (1.01, 1.02]$, TTM $\in [6, 10]$.
- Two groups of BAC call options: $K/S \in (0.98, 1]$, TTM $\in (0, 5]$; $K/S \in (1, 1.02]$, TTM $\in (0, 5]$.

On most of the trading days in 2019, each group contains only one option.

We plot the number of market buy/sell orders for the eight groups of options that are near at the money and short term in one minute intervals from 9:30 a.m. to 4:00 p.m. on one trading day. See figure 2, which clearly demonstrates that the intensity of market orders is unlikely to be constant. In particular, the plots of AAPL options are noticeably U-shaped, indicating that trading is more active around the beginning and the end of the trading day. In comparison, the plots of BAC options are L-shaped. One can also observe clusters of spikes, hinting possible self-excitation.

2.1. Self- and cross-excitation of market orders

In order to detect self-excitation in the buy and sell market orders and cross-excitation between them, we apply the non-parametric approach developed in Bacry *et al.* (2016). We label buy as 0 and sell as 1. To fit into their framework, we assume that the intensity of orders of type i takes the following form for $i = 0, 1$:

$$\lambda^i(t) = \mu^i + \sum_{j\in\{0,1\}} \sum_{t_{jv}<t} \phi^{j,i}(t - t_{jv}), \tag{1}$$

where μ^i is a constant, $\phi^{j,i}(\cdot)$ is the kernel that measures the influence of events of type j on events of type i, and t_{jv} is the occuring time of event v of type j. We can detect self-excitation in buy or sell orders from the kernels $\phi^{0,0}$ or $\phi^{1,1}$ and cross-excitation from $\phi^{0,1}$ and $\phi^{1,0}$.

We used the code provided by Bacry *et al.* (2016) to estimate the kernels with the following data: observations of market buy and sell orders of call options with moneyness in (0.99,1.02] and TTM up to 10 days for AAPL, and call options with moneyness in (0.98,1.02] and TTM up to 5 days for BAC, throughout year 2019. As nonparametric estimation requires a lot of data, we do not estimate the kernels for each of the eight option groups individually and we pool observations from all the trading days in 2019 together for estimation. See figure 3 for the estimation results.

For both stocks, we observe pronounced self-excitation in both buy and sell orders, and the effect is stronger for AAPL as $\phi^{0,0}(t)$ and $\phi^{1,1}(t)$ of AAPL show bigger values for t near zero. However, cross-excitation between buy and sell orders is insignificant compared with their self-excitation as seen from the magnitude of the estimated kernels $\phi^{0,1}$ and $\phi^{1,0}$.

REMARK 1 In general, the baseline intensity μ^i may be time-varying, but we do not consider the general case here as the method of Bacry *et al.* (2016) cannot handle time-varying baseline intensities. This simplification is unlikely to compromise our conclusion on the significance of self-excitation and

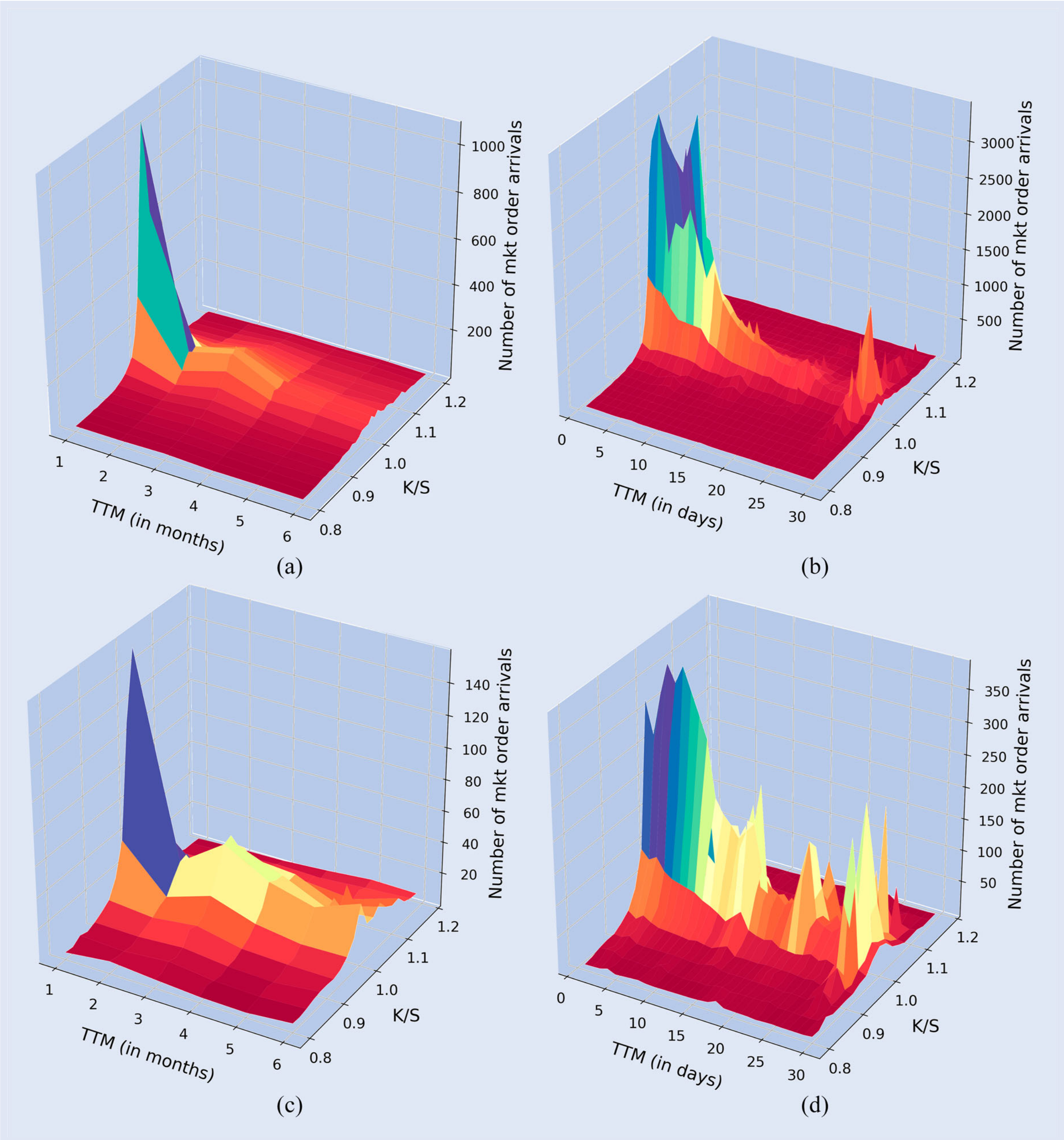

Figure 1. The average daily number of market orders with respect to moneyness and time to maturity. (a) AAPL, TTM < 6 months. (b) AAPL, TTM < 1 month. (c) BAC, TTM < 6 months and (d) BAC, TTM < 1 month.

insignificance of cross-excitation. We emphasize that the nonparametric approach is taken here because we would like to assess the significance of self- and cross-excitation without making specific parametric assumptions. We do not use (1) to generate market order data for training the market making strategy. Based on the nonparametric evidence, we will introduce parametric Hawkes models with time-varying baseline intensities for fitting the data in the next subsection.

2.2. *Estimation of parametric hawkes processes*

The nonparametric approach allows us to understand the significance of self- and cross-excitation without making parametric assumptions. In the following, we would like to estimate the point process of market orders for each option group. We consider parametric Hawkes models instead of the nonparametric one for three reasons. First, nonparametric estimates are typically noisy with limited amount of data. Second, we want the baseline intensity to be time-varying, which is not allowed in the nonparametric approach. Third, we want to know whether the data can be adequately modeled by a Markov point process or not, which cannot be directly seen from the nonparametric estimates.

As cross-excitation is insignificant, we simply assume that there is no cross-excitation in our framework. We model the intensity of market buy orders as

$$\lambda(t) = \mu(t) + \sum_{t_v < t} g(t - t_v), \tag{2}$$

Figure 2. Number of market orders for eight option groups in one minute intervals on the first day when there is an option contract in the group in June 2019. Row 1: AAPL call options with $K/S \in (0.99, 1], (1, 1.01], (1.01, 1.02]$ and TTM up to five days. Row 2: AAPL call options with $K/S \in (0.99, 1], (1, 1.01], (1.01, 1.02]$ and TTM between 6 and 10 days. Row 3: BAC call options with $K/S \in (0.98, 1], (1, 1.02]$ and TTM up to 5 days.

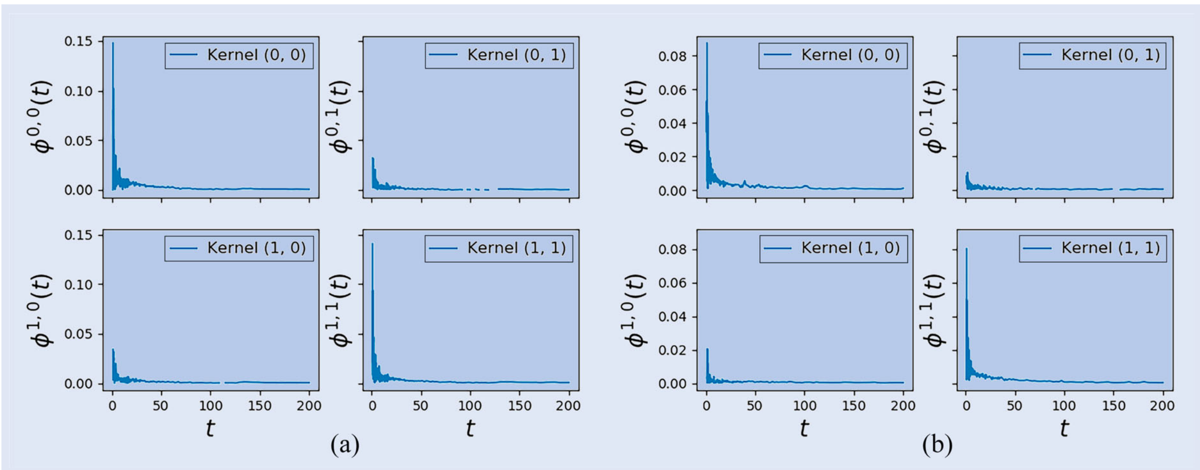

Figure 3. Nonparametric estimation of the kernels for AAPL and BAC call options. (a) AAPL and (b) BAC.

where $\mu(t)$ is a deterministic function representing the baseline intensity, t_v is the arrival time of event v and g is either a power law kernel given by

$$g(t - t_v) = \frac{\alpha}{(t - t_v + \delta)^{\beta}}, \tag{3}$$

or an exponential kernel given by

$$g(t - t_v) = \alpha \mathrm{e}^{-\beta(t - t_v)}. \tag{4}$$

The intensity of market sell orders takes the same form but with different values for the parameters in μ and g. The two parametric kernels considered here are widely used in the Hawkes literature. The Hawkes process is non-Markov under the power law kernel and Markov under the exponential kernel.

In order to fit the U-shaped pattern for AAPL and L-shaped pattern for BAC, we specify $\mu(t)$ as a piecewise function. Specifically, for AAPL $\mu(t)$ is piecewise linear on six time intervals that evenly split the trading day to capture the variations in the number of market orders throughout the day, which corresponds to approximately one hour for each time interval. Let $T_1, \ldots, T_7$ be the endpoints of those intervals and μ_m is the intensity at T_m. The baseline intensity for AAPL is given by

$$\mu(t) = \mu_m + \frac{\mu_{m+1} - \mu_m}{T_{m+1} - T_m} \times (t - T_m),$$
$$T_m \le t \le T_{m+1},\ m = 1, \ldots, 6. \quad (5)$$

For BAC, we could use the same specification, but in simulation trials we find that letting $\mu(t)$ be piecewise constant on three time intervals of equal length can better reproduce the observed pattern for BAC after fitting. This is likely because we have significantly less data for BAC than for AAPL, which makes it difficult to estimate a specification with more parameters accurately. The baseline intensity for BAC is given by

$$\mu(t) = \mu_m, \quad T'_m \le t < T'_{m+1}, \quad m = 1, 2, 3. \quad (6)$$

Table 1. The percentage of samples a model passes the Kolmogorov–Smirnov test.

Underlying	Market order type	Power law kernel Hawkes	Exponential kernel Hawkes	Poisson
AAPL	Buy	91.4%	36.3%	0.094%
	Sell	94.6%	41.9%	0.282%
BAC	Buy	96.3%	83.9%	13.7%
	Sell	97.0%	85.5%	15.5%

Notes: We obtain one sample of X_ks for each option group on each trading day using the estimated model for the group on that day. For each market order type, there is a total of 252 × 6 samples for AAPL options and a total of 252 × 2 samples for BAC options.

On each trading day, we estimate the parameters of two Hawkes models with the power law kernel and the exponential kernel by maximum likelihood estimation using a Python package provided by Takahiro Omi available from GitHub. As a reference, we also estimate a Poisson process with constant intensity. We perform the estimation for the six groups of AAPL call options and for the two groups of BAC call options.

Next, we check the goodness of fit of each model. Let t_k denote the occurring time of the kth event and set $t_0 = 0$. Define $X_k = \int_{t_{k-1}}^{t_k} \lambda(s)ds$ for $k \ge 1$. Then, $X_1, X_2, \ldots$ are i.i.d. and they follow the unit exponential distribution under the true model (Bowsher 2007). Based on the estimated intensity model, we calculate X_ks and apply the Kolmogorov–Smirnov (KS) test to check if they follow the unit exponential distribution at the 5% significance level. The null hypothesis is that the X_ks follow this distribution. If it is not rejected, there is no good reason to think that the estimated intensity model does not provide a good fit to the given data.

We perform the KS test for these three models on each trading day. Table 1 reports the percentage of samples each model can be considered as a good fit according to the KS test. Clearly the Hawkes model with the power law kernel is the best one in all the cases with a high percentage of samples passing the test. The Hawkes model with the exponential kernel also fits the BAC data quite well but not so for the AAPL data. The Poisson model fails for both stocks.

As the Hawkes model with the power law kernel can be considered as a good model for the data, we will use it to generate data for deep learning of the market making strategy. The averages of the parameter estimates of this model over all the trading days in 2019 are reported in table 2. The estimates for μ_ks in the baseline intensity clearly capture the U-shaped pattern for AAPL options. For BAC options, these estimates also reflect the much higher trading activity near the market

Table 2. Average estimates of the Hawkes model parameters.

(a) AAPL

K/S	TTM (days)	Type	μ_1	μ_2	μ_3	μ_4	μ_5	μ_6	μ_7	α	β	δ
(0.99, 1]	0–5	Buy	25.1859	5.5245	3.9878	2.5291	2.8381	2.6329	6.6145	0.0389	1.1969	0.0033
		Sell	25.6986	6.0965	4.5333	2.6307	3.1396	2.8330	6.8661	0.0303	1.1934	0.0520
(0.99, 1]	6–10	Buy	11.5844	3.3777	2.5758	1.7073	2.0056	1.8136	4.5162	0.0199	1.4683	0.0006
		Sell	11.0110	3.4701	2.7819	1.7484	2.0567	1.8686	4.7474	0.0174	1.4833	0.0005
(1, 1.01]	0–5	Buy	32.0986	5.6640	4.8400	2.6295	2.9787	2.8121	9.1477	0.0451	1.0749	0.0003
		Sell	32.7775	6.9804	5.5489	3.1390	3.5938	3.2347	9.5052	0.0366	1.1628	0.0003
(1, 1.01]	6–10	Buy	17.4491	4.1175	3.3483	1.9883	2.0996	1.7884	5.5158	0.0302	1.2405	0.0036
		Sell	16.5548	4.3312	3.5012	1.9861	2.2838	2.0041	5.3050	0.0256	1.4436	0.0004
(1.01, 1.02]	0–5	Buy	24.7130	5.3779	3.8403	2.3960	2.9788	2.1805	7.6171	0.0404	1.1207	0.0013
		Sell	25.1489	6.0735	4.6912	2.6737	3.1682	2.5257	8.1355	0.0320	1.2260	0.0004
(1.01, 1.02]	6–10	Buy	14.6916	3.6668	2.8409	1.5892	1.9844	1.5240	4.9296	0.0540	1.4503	0.0090
		Sell	14.7840	4.0322	3.1479	1.8247	2.2149	1.7988	5.0411	0.0211	1.7351	0.0005

(b) BAC

K/S	TTM (days)	Type	μ_1	μ_2	μ_3	α	β	δ
(0.98, 1]	0–5	Buy	1.8711	0.6952	0.8743	0.0467	1.6872	0.0274
		Sell	1.5729	0.5943	0.7925	0.0662	2.2228	0.0605
(1, 1.02]	0–5	Buy	2.0952	0.7157	0.8984	0.1023	1.6694	0.0859
		Sell	1.8376	0.7294	0.7636	0.0876	2.1879	0.0772

Notes: We estimate the Hawkes parameters on each trading day in 2019 and report the averages of these estimates.

opening. For AAPL options, the ones with a shorter maturity tend to have a larger baseline intensity as seen from the magnitude of μ_ks. In addition, self-excitation persists longer for options with a shorter maturity as their estimated β is smaller. Comparing AAPL and BAC options, self-excitation is less persistent in BAC options with a larger estimated β.

3. Market making for a single option

Consider an agent who does market making for a single option over the time horizon $[0, T]$ at a discrete set of times given by $0 = t_0 < t_1 < t_2 < \cdots < t_K = T$. The option matures after T. At each t_k $(0 \le k \le K-1)$, the agent posts one limit bid order and one limit ask order with their distance to the mid-price given by ϵ_k^b and ϵ_k^a, respectively. For simplicity, we follow Stoikov and Sağlam (2009) and El Aoud and Abergel (2015) to assume that the size of each limit order posted is one option contract although one can easily modify our formulas to allow for an arbitrary but fixed size for the limit order. Baldacci *et al.* (2021) and Baldacci *et al.* (2020) consider a general setting where the agent can post multiple limit orders of different sizes at different price levels. However, in their numerical studies, they assume that each time only one limit order is posted. It is worth noting that although the limit orders are posted only at times on the grid in our setting, they can be executed at any time between two grid points.

The stock price, option price and option delta (the derivative of the option price with respect to the price of the underlying) at t_k are denoted by S_k, C_k and Δ_k, respectively. The prices of the limit bid and ask orders posted time t_k are given by

$$p_k^b = C_k - \epsilon_k^b, \quad p_k^a = C_k + \epsilon_k^a.$$

We denote the numbers of options sold and bought in $[t_k, t_{k+1})$ by $Q_k^a(\epsilon_k^a)$ and $Q_k^b(\epsilon_k^b)$, respectively. Let q_k^o be the option inventory at t_k, and we have

$$q_{k+1}^o = q_k^o + Q_k^b(\epsilon_k^b) - Q_k^a(\epsilon_k^a), \quad k = 0, \ldots, K-1. \tag{7}$$

The stock inventory at t_k is denoted by q_k^s. To hedge the risk of options, we assume the agent applies delta hedging on the time grid. Thus, it holds that

$$q_k^s = -q_k^o \times \Delta_k, \quad k = 0, \ldots, K. \tag{8}$$

We also assume that the stock is very liquid so that we can trade it at price S_k.

Let x_k be the value of the cash the agent holds at t_k immediately after delta hedging is applied. Define $\delta q_{k+1}^s = q_{k+1}^s - q_k^s$, which is the change in the stock inventory from t_k to t_{k+1}. The change in cash from t_k to t_{k+1} after stock position adjustment is given by

$$\begin{aligned} x_{k+1} - x_k = {} & (C_k + \epsilon_k^a) \times Q_k^a(\epsilon_k^a) - (C_k - \epsilon_k^b) \times Q_k^b(\epsilon_k^b) \\ & - \delta q_{k+1}^s S_{k+1}, \quad k = 0, \ldots, K-1. \end{aligned} \tag{9}$$

We can express the agent's initial wealth W_0 and terminal wealth W_K as

$$W_0 = x_0 + q_0^o C_0 + q_0^s S_0, \tag{10}$$

$$\begin{aligned} W_K &= x_K + q_K^o C_K + q_K^s S_K \\ &= x_0 + \sum_{k=0}^{K-1} (x_{k+1} - x_k) \\ &\quad + \left[q_0^o + \sum_{k=0}^{K-1} \left(Q_k^b(\epsilon_k^b) - Q_k^a(\epsilon_k^a) \right) \right] C_K \\ &\quad + \left(q_0^s + \sum_{k=0}^{K-1} \delta q_{k+1}^s \right) S_K. \end{aligned} \tag{11}$$

We consider the exponential utility function given by

$$U(y) = \begin{cases} \dfrac{1 - e^{-\gamma y}}{\gamma}, & \gamma > 0, \\ y, & \gamma = 0, \end{cases}$$

where the parameter γ measures the degree of risk-aversion. The agent is risk-neutral if $\gamma = 0$ and risk-averse if $\gamma > 0$. This utility function is a popular choice in the literature (see e.g.Avellaneda and Stoikov 2008, Baldacci *et al.* 2021). We consider maximizing the expected utility of the market making profit, i.e.

$$\max_{\substack{\epsilon_k^a, \epsilon_k^b > 0 \\ k = 0, \ldots, K-1}} \mathrm{E}[U(W_K - W_0)]. \tag{12}$$

For the exponential utility, since $e^{-\gamma W_K} = e^{-\gamma W_0} e^{-\gamma (W_K - W_0)}$, we have for $\gamma > 0$,

$$\begin{aligned} \text{argmax } \mathrm{E}[U(W_K)] &= \text{argmax } \mathrm{E}[-e^{-\gamma W_K}] \\ &= \text{argmax } \mathrm{E}[-e^{-\gamma (W_K - W_0)}] \\ &= \text{argmax } \mathrm{E}[U(W_K - W_0)], \end{aligned}$$

where argmax refers to the optimal strategy. Thus, the formulation (12) can be viewed as equivalent to maximizing the expected utility of the terminal wealth in terms of the optimal strategy. In this section, we will not consider constraints for simplicity, which will be dealt with in Section 5.

3.1. *Market orders and execution of limit orders*

We model the arrivals of market buy and sell orders by the Hawkes processes with a time-varying baseline intensity given by either (5) or (6) and the power law kernel given by (3).

The probability of a limit order posted at t_k executed in the time interval $[t_k, t_{k+1})$ depends on the distance of the limit price to the mid-price and the number of market orders in this time interval. A market order can eat into different levels of limit orders and thus generates market impact. It is typical practice in the literature to assume a specific form for the market impact function. The deep learning approach can handle general forms of the market impact function, but to carry out numerical experiments, we need to assume a specific form. We follow Avellaneda and Stoikov (2008) and assume that the size Q of a market order follows the power law distribution $f^Q(x) \propto x^{-1-\nu}$ for some $\nu > 0$. Let p^Q be the price of the farthest limit order executed in the trade with a market order of size Q, and consider its distance to mid-price $\delta p := p^Q - p^{\text{mid}}$. As in Avellaneda and Stoikov (2008), we assume $\delta p \propto \ln(Q)$, which is supported by the empirical findings in Potters and Bouchaud (2002). Under this assumption, the probability of a limit order with ϵ as the distance of its price to the mid-price is given by

$$\Pr(\delta p > \epsilon) = \mathrm{e}^{-\kappa\epsilon}, \tag{13}$$

for some constant $\kappa > 0$. Let $N^{mkt}_{j,j+1}$ be the number of market orders arrived in $[t_j, t_{j+1})$ and $Q(\epsilon_j)$ be the execution status for the limit order posted at t_j with ϵ_j as the distance to the mid-price. We label execution as 1 and the other case as 0. Then the probability of the limit order executed in $[t_j, t_{j+1})$ can be calculated as follows:

$$\begin{aligned}\Pr(Q(\epsilon_j) = 1|N^{mkt}_{j,j+1}) &= 1 - \Pr(Q(\epsilon_j) = 0|N^{mkt}_{j,j+1}) \\ &= 1 - \prod_{i=1}^{N^{mkt}_{j,j+1}} \left(1 - \Pr(\delta p > \epsilon_j)\right) \\ &= 1 - \left(1 - \mathrm{e}^{-\kappa\epsilon_j}\right)^{N^{mkt}_{j,j+1}}. \end{aligned} \tag{14}$$

In this calculation, for simplicity we assume that the limit order book is resilient and restores its previous shape after a market order is executed. Thus, (14) is only an approximation of the true execution probability.

REMARK 2 Ideally, when the mid-price changes during $[t_j, t_{j+1})$, we update the distance to the mid-price before the limit order gets executed. However, we are not able to do this with the CBOE data, which provides the stock and option prices at one minute frequency. This is also the market making frequency we chose for the AAPL options. In any one-minute time interval, we are not able to observe the change in mid-price from the data.

3.2. *Neural network approximation*

To solve the stochastic control problem (12), we apply the deep learning approach developed in Han and Weinan (2016), which turns a dynamic control problem into a stochastic optimization problem. For $k = 0, \ldots, K-1$, we approximate ϵ^a_k and ϵ^b_k each by a feedforward neural network (FNN); see figure 4 for an illustration. The neural networks for ϵ^a_k and ϵ^b_k share the same structure, but with different inputs and parameter values. In total, we have $2K$ neural networks for the control variables $\epsilon^a_k, \epsilon^b_k, k = 0, \ldots, K-1$.

In our implementation, we use the following specification for each neural network: two hidden layers with 64 neurons on each layer, $\mathrm{ReLU}(x) = \max(x, 0)$ as the activation function of the hidden layers and $\mathrm{Sigmoid}(x) = 1/(1 + \exp(-x))$ as the activation function of the output layer. Using the sigmoid function implies that we place the limit orders within a dollar distance of the mid-price. In reality, the CBOE adopts the following tick rule which applies to AAPL and BAC: \$0.01 for options trading below \$3 and \$0.05 for options trading at or above \$3. The prices of AAPL and BAC options under our consideration are typically between \$0.5 to \$5 and \$0.15 to \$2, respectively. Thus, the maximum one dollar distance allowed by the sigmoid function is equivalent to placing the limit orders at most 100 ticks or 20 ticks away from the mid-price.

REMARK 3 If a general range is needed for ϵ^a_k or ϵ^b_k, we can use the generalized sigmoid function $\mathrm{GSigmoid}(x) = \frac{A}{1+\mathrm{e}^{-x}} + B$ $(A > 0)$ to activate the output layer, which produces the range $(B, B + A)$. When B is negative, the distance to the mid-price from the neural network can be negative. In such case, the market maker posts a market order instead of a limit order.

We use the state variables $(S_k, C_k, x_k, q^o_k, q^s_k)$ plus another feature from the Hawkes process as inputs to the neural networks. Specifically, the input vector for the FNN of ϵ^a_k is $(S_k, C_k, x_k, q^o_k, q^s_k, f_{b,k})$ and the input vector for the FNN of

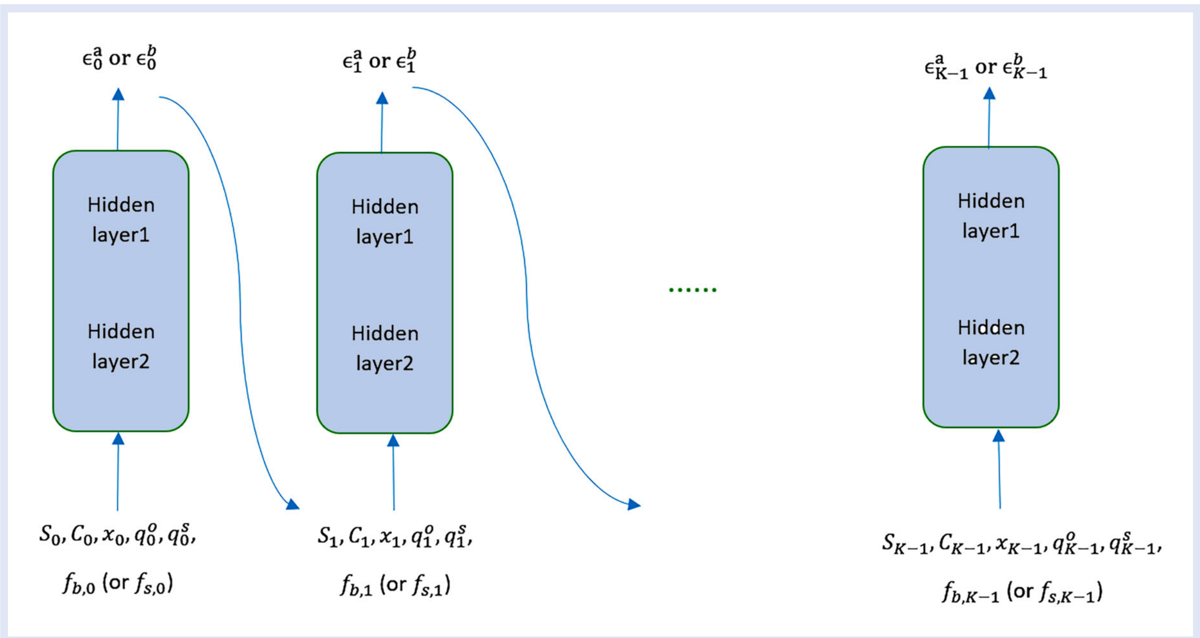

Figure 4. Illustration of the network architecture.

ϵ_k^a is $(S_k, C_k, x_k, q_k^o, q_k^s, f_{s,k})$. Here, $f_{b,k}$ and $f_{s,k}$ are features related to the number of market buy and sell orders in the next period $[t_k, t_{k+1})$, respectively. With more market orders in $[t_k, t_{k+1})$, the execution probability of limit orders becomes higher. Thus, the agent can place the limit orders further away from the mid-price to gain a larger spread if she expects more market orders will arrive.

We consider two specifications for $f_{b,k}$ and $f_{s,k}$ that are potentially informative to the neural network for the number of market orders in $[t_k, t_{k+1})$.

(1) The intensity of the Hawkes process at t_k is used as the feature, i.e.

$$f_{b,k} = \lambda^0(t_k), \quad f_{s,k} = \lambda^1(t_k),$$

where 0 and 1 refer to market buy and sell orders, respectively, and

$$\lambda^i(t_k) = \mu^i(t_k) + \sum_{t_v^i < t_k} \frac{\alpha^i}{(t_k - t_v^i + \delta^i)^{\beta^i}}.$$

The intensity at t_k indicates the probability of an arrival in a very short period of time starting at t_k.

(2) Let $N_{k,k+1}^i$ be the number of market orders of type i arrived in $[t_k, t_{k+1})$, and $\mathcal{F}_k^i$ is the information generated by the market order process of type i up to time t_k. We set

$$f_{b,k} = E\left[N_{k,k+1}^0 | \mathcal{F}_k^0\right], \quad f_{s,k} = E\left[N_{k,k+1}^1 | \mathcal{F}_k^1\right].$$

One can view these conditional expectations as the predicted number of market buy and sell orders in the next time interval, which is a natural feature for obtaining the market making strategy at t_k. To calculate these expectations, we transform the problem to fit the setting in Gao *et al.* (2018) and apply their result.

For $t \in [t_k, t_{k+1})$, the intensity is given by

$$\lambda^i(t) = \mu^i(t) + \sum_{t_v^i \le t_k} \frac{\alpha^i}{(t - t_v^i + \delta^i)^{\beta^i}} + \sum_{t_k < t_v^i < t} \frac{\alpha^i}{(t - t_v^i + \delta^i)^{\beta^i}}.$$

Let $u = t - t_k$. We rewrite the intensity as a function of u as the following:

$$\begin{aligned} \lambda_k^i(u) &= \mu^i(t) + \sum_{t_v^i \le t_k} \frac{\alpha^i}{(t_k + u - t_v^i + \delta^i)^{\beta^i}} \\ &\quad + \sum_{t_k < t_v^i < t_k + u} \frac{\alpha^i}{(t_k + u - t_v^i + \delta^i)^{\beta^i}} \\ &= \nu_k^i(u) + \sum_{0 < t_v^i - t_k < u} h_k^i\left(u - (t_v^i - t_k)\right), \end{aligned}$$

where

$$\nu_k^i(u) = \mu^i(t_k + u) + \sum_{t_v^i \le t_k} \frac{\alpha^i}{(t_k + u - t_v + \delta^i)^{\beta^i}},$$

$$h_k^i(x) = \frac{\alpha^i}{(x + \delta^i)^{\beta^i}}.$$

By Proposition 5 in Gao *et al.* (2018), the expected number of events in $[t_k, t_{k+1})$ is given by

$$E\left[N_{k,k+1}^i | \mathcal{F}_k^i\right] = \int_0^{t_{k+1} - t_k} \nu_k^i(t_{k+1} - t_k - u)\psi_k^i(u)\, du, \quad (15)$$

where $\psi_k^i(u)$ is the unique solution to the Volterra integral equation

$$\psi_k^i(u) = 1 + \int_0^u h_k^i(u - s)\psi_k^i(s)\, ds. \quad (16)$$

We solve (16) by the finite difference method with 2000 grid points and then calculate the expected number of market orders from (15) by the Gauss–Legendre quadrature.

Remark 4 The deep learning method developed in Han and E (2016) for solving stochastic control problems approximates controls of different times by different neural networks. This method has been successfully applied to solve the hedging problem in two frameworks in Buehler *et al.* (2019a) and Carbonneau and Godin (2021), where the authors also view it as a special deep reinforcement learning algorithm. It is also called deep empirical risk minimization in Reppen and Soner (2022). Other neural network structures can certainly be used to solve a control problem. We can approximate the controls of different times by a single FNN with the decision time as an additional input to the network. This single FNN structure is employed in Fecamp *et al.* (2020) for hedging. One can also utilize a recursive neural network (RNN) to output the control (see Zhang and Huang 2021 for hedging) or combine an RNN with an FNN to obtain the control (the output of an RNN is used as a feature for the FNN which outputs the control; see Buehler *et al.* 2019b for hedging). A detailed comparison of these neural network structures is provided in Dai *et al.* (2022) for the hedging problem. We do not consider other structures for solving the market making problem for the following reasons. First, the multiple FNN structure proposed in Han and E (2016) provides more flexibility than the single FNN structure. Although the latter may be easier and faster to train when there are many periods in the control problem, we only consider market making over a relatively short time horizon, so there is no need to sacrifice flexibility for computational efficiency. Second, since the process of option market order arrivals is non-Markov, it is natural to consider combining an RNN with an FNN. We can use the RNN to predict the number of market orders in $[t_k, t_{k+1})$, which can be used as a feature for the FNN at t_k to output the quote. Nevertheless, an RNN involves substantially more parameters than the parametric Hawkes model in Section 2.2 that fits the data well, which makes it prone to overfit and likely to perform worse out of sample. Alternatively, one may be tempted to apply value-based reinforcement learning algorithms like Q-learning to solve our problem. However, as the Hawkes process for option market order arrivals is non-Markovian in light of our empirical results, the value function at a given time depends on the whole path of market orders up to that time, making it difficult to apply value-based reinforcement learning algorithms.

3.3. *The deep learning algorithm*

Let Θ denote the vector of all the parameters of the $2K$ FNNs. The deep learning algorithm requires N simulated paths with the market making profit and loss (PnL) on each path for a given Θ. The procedure consists of the following steps.

(1) We generate N paths of stock prices, option prices and deltas by simulation. On each path, we first simulate stock prices on the time grid $\{0, t_1, \ldots, t_K\}$ and then calculate the corresponding option prices and deltas. We also simulate the arrival times of market buy and sell orders between time 0 and T according to the Hawkes process with a time-varying baseline and the power law kernel.

(2) To obtain the market making PnL on each path for a given Θ, repeat the following steps for $k = 0, \ldots, K-1$: (1) At t_k, apply the bid and ask quotes from the neural networks with the given Θ. Simulate the execution of the limit orders in $[t_k, t_{k+1}]$ based on the simulated market order arrivals in this time interval and the execution probability given by (14). (2) Calculate $q^o_{k+1}, q^s_{k+1}, x_{k+1}$ by (7), (8), (9), respectively. Finally, we calculate $W_K - W_0$ with W_0 and W_K given by (10) and (11).

Since the terminal wealth depends on the neural network parameters, we will write it as $W_k(\Theta)$. We now approximate $\mathrm{E}[U(W_K(\Theta) - W_0)]$ using the sample average:

$$\mathrm{E}[U(W_K(\Theta) - W_0)] \approx \frac{1}{N}\sum_{j=1}^{N} U(W_K^j(\Theta) - W_0).$$

where W_K^j is the terminal wealth on path j. Consider the loss function

$$\ell(\Theta) = -\frac{1}{N}\sum_{j=1}^{N} U(W_K^j(\Theta) - W_0).$$

The original stochastic control problem (12) now becomes a stochastic optimization problem as follows:

$$\min_{\Theta} \ell(\Theta).$$

We will solve this problem by the stochastic gradient descent method with the Adam optimizer (Kingma and Ba 2014). We also use mini-batches to reduce computation cost. Specifically, the full dataset is randomly partitioned into mini-batches. In each iteration, we only use the paths in one mini-batch to update the gradient of the loss function.

REMARK 5 Step 1 of the algorithm requires a stock price model for simulation and option pricing. The deep learning approach can handle general and sophisticated models, such as stochastic volatility models, as long as one can simulate them and price options.

REMARK 6 Since the outcome of the execution of a limit order is binary, the loss function $\ell(\Theta)$ is not differential in Θ. To restore differentiability, in our implementation we use the relaxed Bernoulli distribution, which is a continuous distribution over [0, 1], to approximate the Bernoulli distribution.

4. Empirical results for single option market making

We assume the time horizon for market making is one day and post limit orders at one minute frequency for AAPL (390 periods in one day) and at five minute frequency for BAC (78 periods in one day). In reality, the market making frequency could be higher, but we have to align it with the frequency of the CBOE data which is one minute for the out-of-sample test. For BAC, we consider five minutes because its market orders are less frequent. We solve the market making problem for a risk-neutral agent with $\gamma = 0$ and a risk-averse agent with $\gamma = 0.05$.

We evaluate the performances of six market making strategies summarized in table 3. In this section, the Hawkes process always refers to the one with the time-varying baseline (equation (5) for AAPL and equation (6) for BAC) and the power law kernel given by (3). Depending on the model that generates the market orders for training the neural networks and the feature as an additional input to the neural networks besides the state variables, we have four strategies from deep learning. The Poisson strategy is obtained by the deep learning algorithm with market orders generated by the Poisson process. Since the arrival rate is constant, there is no feature to use for the neural networks besides the state variables. We also consider another two strategies. LIF is a looking-into-future strategy obtained by the deep learning algorithm with market orders generated by the Hawkes process. We use the exact number of market orders in $[t_k, t_{k+1})$ as an input for the neural network at t_k in addition to the state variables. This allows the market maker to peek into the future to optimize her quotes, which is definitely better than strategies based only on information up to t_k. One can view the non-adapted LIF strategy as an unattainable upper bound for the adapted deep learning strategies. Strategy Constant is obtained without using deep learning. We simply assume ϵ^a and ϵ^b remain constant over time (but they can be different). We specify them as multiples of the tick size (set to \$0.01) and search the grid up to 100 ticks to find the ϵ^a and ϵ^b that maximize the expected utility approximated by its sample average. The market order data is again generated by the Hawkes process.

Table 3. Description of six market making strategies.

Name	Generation Model	Additional Input for the Neural Networks
LIF	Hawkes	the exact number of market orders in the next interval
ExpN	Hawkes	expected number of market orders in the next interval
Intensity	Hawkes	the Hawkes intensity
NoF	Hawkes	none
Poisson	Poisson	none
Constant	Hawkes	not applicable

4.1. The experiment design

We do training and testing on a rolling basis using the data in 2019 for six groups of AAPL call options and two groups of BAC call options as specified in Section 2. On most of the trading days, there is only one option contract in each group. Trading at CBOE goes from 9:30 a.m. to 16:00 p.m. We drop the trading days which lasted less than 6.5 hours, and 248 days are left for our consideration.

For $n = 1, \ldots, 11$, we use the CBOE data of month n to estimate the Hawkes parameters and generate the market order data to obtain the market making strategies, and then use the CBOE data of month $n + 1$ to test these strategies. In total, there are 11 months of data to evaluate them. On each trading day, we estimate the Hawkes parameters of each option group and check the goodness of fit by the Kolmogorov-Smirnov (KS) test. So there is a set of parameters for each option group on each day. For month n, we retain only those sets of parameters that pass the KS test, the percentage of which is very high as seen from table 1. To train the neural networks for market making or searching for the optimal constant strategy, we simulate 1000 daily paths of market orders from each accepted set of Hawkes parameters, which amounts to roughly 15, 000 to 20, 000 daily paths for the whole month. Paths generated from different sets will be used, thus parameter uncertainty in the Hawkes process is factored into our solutions of the market making problem. When the Poisson process is used for the market orders, we follow the same procedure except that the KS test is not applied to check the goodness of fit.

As for the stock price model, since our experiment requires repeated model estimation and option pricing, we simply use the Black-Scholes (BS) model for tractability, i.e.

$$\mathrm{d}S_t = \mu S_t \,\mathrm{d}t + \sigma S_t \,\mathrm{d}B_t.$$

Stochastic volatility models are more realistic choices, but the stochastic volatility effect would be small for the very short-term options considered in the paper. Furthermore, all options we tested are near the money, so there is also less need to produce volatility smiles or skews by stochastic volatility.† For the simulation in month n ($n = 1, \ldots, 11$), we first estimate μ and σ based on the daily stock returns in the preceding 11 months and month n and then simulate 100 000 daily stock price paths from the estimated model. The initial stock price is generated from $[160, 290]$ for AAPL and $[25, 35]$ for BAC, which cover their price ranges in 2019. We then calculate the option price and delta on each path. For the risk-free rate, we use the one-month US Treasury bill rate. For each option group, we set the strike of the option in that group using the midpoint of its moneyness range.

Given a market making strategy, to simulate the execution of limit orders, we set the price impact parameter $\kappa = 10$ in (13). If there is only one market order in a time period, it is easy to show that to maximize the expected profit of executing one limit order with the execution probability given by (13), the optimal distance to the mid-price is given by $1/\kappa$. It equals \$0.1 with $\kappa = 10$, which is reasonable given the tick size is either \$0.01 or \$0.05 for the AAPL and BAC options.

For each option group, we perform training and testing for $n = 1, \ldots, 11$ as follows:

- Training: in month n, to solve the one-day market making problem, we randomly pair the daily paths of stock price, option price and delta with the daily paths of market orders to obtain a total of 100, 000 daily paths. We use 80% of them for training the neural networks and 20% for validation. We also use the same training paths to find the optimal constant strategy. As for the initial condition of market making, we always set the initial cash value $x_0 = 5000$, and the initial option inventory $q_0^o = 10$.
- Testing: we apply every market making strategy obtained from training on each trading day of month $n + 1$ using the the stock price, option price, delta and market orders in the CBOE data. We simulate 500 paths of uniform random variables on $[0, 1]$ on each trading day to simulate the execution of limit orders. We adopt the common random number technique, which uses the same paths of uniform random variables for different strategies, to reduce the variance of estimating their differences. On each trading day, we have 500 realized market making profits for every strategy.

We collect the realized market making profits from all the test days, which gives us a sample of roughly 110 000 observations for every strategy.

Remark 7 It would be more realistic to model the option prices and market orders jointly. In our framework which follows from Avellaneda and Stoikov (2008), we can use the ideas in e.g. Cartea *et al.* (2014), Bacry and Muzy (2014), and Fodra and Pham (2015) to introduce dependence, but some inconsistencies can still occur as pointed out by Law and Viens (2019). Alternatively, we can model the limit order book of options as in Law and Viens (2019) for equity market making. However, we do not have data to calibrate such model for out-of-sample testing and there exist challenges in estimating such a complicated model. Nevertheless, this is an important direction to pursue in future research. It is difficult to know how much the empirical performance of our market making strategies would be affected if the joint behavior were not considered. But we expect that the market making profits would be lower than the figures reported in our study. Nevertheless, our deep learning approach is flexible and one can apply a joint model to generate data to train the neural networks when it becomes available.

† We tried Heston's stochastic volatility model with the same data on short-term options and the results can be found in Lai (2021). In the out-of-sample test, using the Heston model produces inferior market making results under all strategies than using the BS model. The Heston model does not provide a good fit to the option prices and deltas out of sample, i.e. the option prices and deltas produced by the Heston model fitted to the training dataset do not seem to be close enough to those on the test dataset. The BS model, albeit simple, is a more robust choice for short-term options when they are near the money.

4.2. Convergence of the deep learning algorithm

We minimize the loss function using the stochastic gradient descent method with minibatches. The size of a minibatch is 128 and we iterate until all the minibatches have been used. We use the Adam optimizer and set the learning rate as follows. For AAPL, the learning rate $LR = 0.005$ for the first half iterations, and $LR = 0.001$ for the second half iterations for $\gamma = 0$; $LR = 0.001$ for $\gamma = 0.05$. For BAC, $LR = 0.001$ for the first half iterations, and $LR = 0.0002$ for the second half iterations for both $\gamma = 0$ and $\gamma = 0.05$. We illustrate the convergence of the deep learning algorithm in figure 5 for two cases. It can be observed that the algorithm converges quite fast. In both cases, it shows convergence in less than 100 iterations. The training loss exhibits oscillations, which is due to the randomness in estimating the loss gradient with minibatches. There is also no sign of overfitting, as the validation loss is close to the training loss. The out-of-sample test loss is also close to them.

4.3. Findings

Based on the realized market making profits from all the test days, we calculate the average daily profit for the risk-neutral agent and the average daily utility for the risk-averse agent. See table 4 for AAPL and table 5 for BAC. To compare different strategies and see if their differences are statistically significant, we also construct a 99% confidence interval for the difference in the expected daily profit or utility for each pair of strategies, and they are displayed in tables 6–8. These tables should be read as follows. Each entry considers the expected daily profit/utility of the row strategy minus that of the column strategy. If the interval does not include zero and it is on the positive/negative real line for a pair of strategies, the difference is positive/negative at the 1% significance level. In tables 9–11, we also show several statistics of the PnL distribution of these strategies. Furthermore, figures 6 and 7 plot the strategies for one day in 2019, and figure 8 shows the inventory of the same options on the same day. Note that common

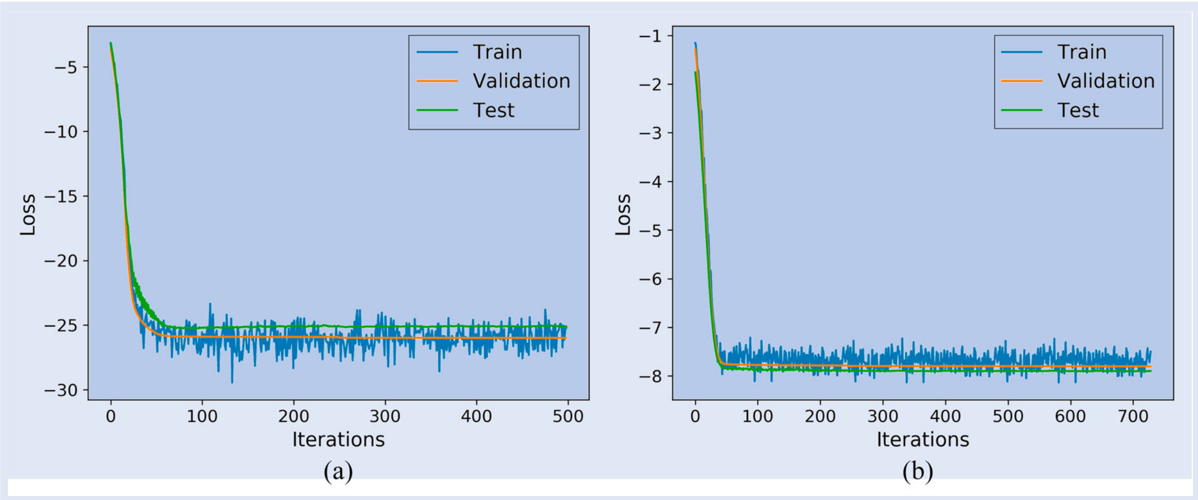

Figure 5. Convergence of the loss function for the market making problem of the risk-neutral agent. Plot (a) considers the AAPL option group $K/S \in (1.01, 1.02]$, TTM $\in [6, 10]$. The training loss and validation loss are calculated based on the training and validation datasets of January 2019, and the test loss is calculated using the market data in February 2019. Plot (b) considers the BAC option group $K/S \in (1, 1.02]$, TTM $\in (0, 5]$. The training loss and validation loss are calculated based on the training and validation datasets of May 2019, and the test loss is calculated using the market data in June 2019. (a) AAPL and (b) BAC.

Table 4. Average daily profit/utility of AAPL options on the test data.

K/S	(0.99, 1]		(1, 1.01]		(1.01, 1.02]	
TTM (days)	0–5	6–10	0–5	6–10	0–5	6–10
			(a) $\gamma = 0$			
LIF	41.0804	19.9435	55.3756	30.6304	48.1449	28.3898
ExpN	39.0396	18.9734	52.6331	29.3628	45.5686	27.1647
Intensity	39.0582	18.9901	52.5518	29.3201	45.6146	27.2055
NoF	38.7620	19.0000	52.2596	29.2700	45.0160	27.0329
Poisson	37.2899	17.7188	51.2257	28.3030	44.0804	26.1534
Constant	38.5745	18.4307	51.4311	29.1097	44.8105	27.0419
			(b) $\gamma = 0.05$			
LIF	16.3153	11.2406	18.4140	14.7048	17.4690	14.1905
ExpN	16.1011	11.0015	18.2569	14.4774	17.2513	13.9721
Intensity	16.1038	10.9995	18.2530	14.4712	17.2483	13.9727
NoF	16.0571	10.8991	18.2198	14.4067	17.1835	13.9191
Poisson	15.6369	10.4336	17.9033	14.0530	17.0341	13.5979
Constant	16.0371	10.8901	18.1561	14.3928	17.1244	13.7611

Table 5. Average daily profit/utility of BAC options on the test data.

K/S	(0.98, 1]	(1, 1.02]
	(a) $\gamma = 0$	
LIF	6.0764	8.0787
ExpN	5.8036	7.7385
Intensity	5.8021	7.7290
NoF	5.7512	7.6177
Poisson	5.5945	7.5197
Constant	5.7664	7.6190
	(b) $\gamma = 0.05$	
LIF	5.0290	6.4960
ExpN	4.8436	6.2820
Intensity	4.8396	6.2722
NoF	4.8092	6.2097
Poisson	4.6976	6.1493
Constant	4.8128	6.2116

random numbers are used in simulating the execution results of different strategies.

We first compare two strategies: NoF and Poisson. The former uses the Hawkes model for simulating the market order arrivals, which is more realistic than the simple Poisson process. Both of them do not use any feature derived from the market order process, so their difference reflects the amount of improvement switching to a more realistic model alone brings. We observe that for all option groups, the average daily profit/utility of NoF is greater than the one of Poisson at the 1% significance level. Moreover, from figures 6 and 7, the optimal quotes throughout the day stay relatively constant for Poisson, while they are noticeably much higher for NoF near the market opening. Since trading is more active in the initial hours for both AAPL and BAC options, one should place limit orders at a greater distance to the mid-price to take advantage of it. However, a model with constant intensity cannot produce the desired strategy. This comparison highlights the importance of modeling the option market orders in a realistic way.

We then check whether using any feature from the Hawkes model can further improve the performance by comparing three strategies: NoF, Intensity and ExpN. Except for one option group of AAPL options for the risk-neutral agent, Intensity and ExpN demonstrate improvement over NoF at the 1% significance level. In the exception, the differences among them are actually very small. The improvement is

Table 6. 99% confidence intervals of the differences in the average daily profits of several strategies for AAPL options ($\gamma = 0$).

	Intensity	NoF	Poisson	Constant
		(a) $0.99 < K/S \leq 1$, TTM ≤ 5 days		
ExpN	(−0.0205, −0.0168)	(0.2747,0.2805)	(1.7444,1.7550)	(0.4617,0.4685)
Intensity		(0.2933,0.2990)	(1.7632,1.7735)	(0.4802,0.4873)
NoF			(1.4669,1.4774)	(0.1843,0.1908)
Poisson				(−1.2903, −1.2788)
		(b) $0.99 < K/S \leq 1$, $6 \leq$ TTM ≤ 10 days		
ExpN	(−0.0180, −0.0154)	(−0.0286, −0.0245)	(1.2501,1.2591)	(0.5351,0.5503)
Intensity		(−0.0119, −0.0079)	(1.2668,1.2758)	(0.5518,0.5670)
NoF			(1.2765,1.2859)	(0.5611,0.5775)
Poisson				(−0.7185, −0.7052)
		(c) $1 < K/S \leq 1.01$, TTM ≤ 5 days		
ExpN	(0.0793,0.0834)	(0.3708,0.3762)	(1.4009,1.4139)	(1.1983,1.2058)
Intensity		(0.2894,0.2948)	(1.3196,1.3325)	(1.1170,1.1243)
NoF			(1.0277,1.0402)	(0.8243,0.8328)
Poisson				(−0.2117, −0.1991)
		(d) $1 < K/S \leq 1.01$, $6 \leq$ TTM ≤ 10 days		
ExpN	(0.0411,0.0443)	(0.0905,0.0951)	(1.0550,1.0646)	(0.2501,0.2559)
Intensity		(0.0478,0.0524)	(1.0124,1.0218)	(0.2075,0.2132)
NoF			(0.9628,0.9712)	(0.1571,0.1633)
Poisson				(−0.8114, −0.8021)
		(e) $1.01 < K/S \leq 1.02$, TTM ≤ 5 days		
ExpN	(−0.0479, −0.0441)	(0.5496,0.5555)	(1.4797,1.4967)	(0.7549,0.7615)
Intensity		(0.5956,0.6015)	(1.5258,1.5426)	(0.8009,0.8074)
NoF			(0.9268,0.9444)	(0.2019,0.2093)
Poisson				(−0.7386, −0.7215)
		(f) $1.01 < K/S \leq 1.02$, $6 \leq$ TTM ≤ 10 days		
ExpN	(−0.0423, −0.0392)	(0.1295,0.1341)	(1.0068,1.0157)	(0.1159,0.1297)
Intensity		(0.1702,0.1750)	(1.0476,1.0566)	(0.1565,0.1706)
NoF			(0.8753,0.8837)	(−0.0150, −0.0030)
Poisson				(−0.8962, −0.8809)

Notes: Each entry considers the expected daily profit of the row strategy minus that of the column strategy. If the interval does not include zero and it is on the positive/negative real line for a pair of strategies, the difference is positive/negative at the 1% significance level.

Table 7. 99% confidence intervals of the differences in the average daily utilities of several strategies for AAPL options ($\gamma = 0.05$).

	Intensity	NoF	Poisson	Constant
		(a) $0.99 < K/S \leq 1$, TTM ≤ 5 days		
ExpN	(−0.0030, −0.0024)	(0.0435,0.0444)	(0.4629,0.4654)	(0.0632,0.0647)
Intensity		(0.0462,0.0471)	(0.4656,0.4681)	(0.0659,0.0673)
NoF			(0.4188,0.4216)	(0.0193,0.0207)
Poisson				(−0.4016, −0.3988)
		(b) $0.99 < K/S \leq 1$, $6 \leq$ TTM ≤ 10 days		
ExpN	(0.0016,0.0025)	(0.1017,0.1031)	(0.5659,0.5699)	(0.1093,0.1136)
Intensity		(0.0996,0.1011)	(0.5639,0.5678)	(0.1072,0.1116)
NoF			(0.4635,0.4675)	(0.0068,0.0113)
Poisson				(−0.4602, −0.4528)
		(c) $1 < K/S \leq 1.01$, TTM ≤ 5 days		
ExpN	(0.0038,0.0041)	(0.0368,0.0373)	(0.3525,0.3548)	(0.1005,0.1012)
Intensity		(0.0329,0.0334)	(0.3486,0.3508)	(0.0965,0.0973)
NoF			(0.3154,0.3178)	(0.0634,0.0641)
Poisson				(−0.2539, −0.2517)
		(d) $1 < K/S \leq 1.01$, $6 \leq$ TTM ≤ 10 days		
ExpN	(0.0059,0.0066)	(0.0702,0.0712)	(0.4227,0.4262)	(0.0837,0.0855)
Intensity		(0.0640,0.0650)	(0.4165,0.4200)	(0.0775,0.0793)
NoF			(0.3519,0.3556)	(0.0130,0.0148)
Poisson				(−0.3419, −0.3378)
		(e) $1.01 < K/S \leq 1.02$, TTM ≤ 5 days		
ExpN	(0.0017,0.0042)	(0.0664,0.0692)	(0.2139,0.2205)	(0.1251,0.1287)
Intensity		(0.0633,0.0664)	(0.2106,0.2179)	(0.1220,0.1258)
NoF			(0.1456,0.1531)	(0.0572,0.0610)
Poisson				(−0.0943, −0.0863)
		(f) $1.01 < K/S \leq 1.02$, $6 \leq$ TTM ≤ 10 days		
ExpN	(−0.0010, −0.0003)	(0.0525,0.0535)	(0.3724,0.3759)	(0.2086,0.2133)
Intensity		(0.0531,0.0542)	(0.3730,0.3766)	(0.2093,0.2139)
NoF			(0.3195,0.3228)	(0.1556,0.1604)
Poisson				(−0.1668, −0.1596)

Notes: Each entry considers the expected daily utility of the row strategy minus that of the column strategy. If the interval does not include zero and it is on the positive/negative real line for a pair of strategies, the difference is positive/negative at the 1% significance level.

Table 8. 99% confidence intervals of the differences in the average daily profits/utilities of several strategies for BAC options.

	Intensity	NoF	Poisson	Constant
		(a) $0.98 < K/S \leq 1$, TTM ≤ 5 days, $\gamma = 0$		
ExpN	(0.0012,0.0018)	(0.0521,0.0529)	(0.2085,0.2099)	(0.0368,0.0377)
Intensity		(0.0506,0.0514)	(0.2070,0.2084)	(0.0353,0.0362)
NoF			(0.1561,0.1573)	(−0.0157, −0.0149)
Poisson				(−0.1727, −0.1712)
		(b) $1 < K/S \leq 1.02$, TTM ≤ 5 days, $\gamma = 0$		
ExpN	(0.0090,0.0099)	(0.1202,0.1214)	(0.2179,0.2197)	(0.1188,0.1202)
Intensity		(0.1107,0.1120)	(0.2084,0.2103)	(0.1093,0.1107)
NoF			(0.0971,0.0989)	(−0.0020, −0.0007)
Poisson				(−0.1002, −0.0985)
		(c) $0.98 < K/S \leq 1$, TTM ≤ 5 days, $\gamma = 0.05$		
ExpN	(0.0039,0.0043)	(0.0342,0.0347)	(0.1456,0.1465)	(0.0306,0.0312)
Intensity		(0.0301,0.0306)	(0.1415,0.1424)	(0.0265,0.0271)
NoF			(0.1112,0.1120)	(−0.0038, −0.0033)
Poisson				(−0.1157, −0.1147)
		(d) $1 < K/S \leq 1.02$, TTM ≤ 5 days, $\gamma = 0.05$		
ExpN	(0.0096,0.0101)	(0.0719,0.0727)	(0.1322,0.1333)	(0.0700,0.0709)
Intensity		(0.0620,0.0629)	(0.1223,0.1234)	(0.0602,0.0610)
NoF			(0.0598,0.0610)	(−0.0023, −0.0015)
Poisson				(−0.0629, −0.0617)

Notes: Each entry considers the expected daily profit/utility of the row strategy minus that of the column strategy. If the interval does not include zero and it is on the positive/negative real line for a pair of strategies, the difference is positive/negative at the 1% significance level.

Table 9. Statistics of the daily PnL distribution for AAPL options with $\gamma = 0$.

K/*S*	TTM (days)	strategy	1% quantile	99% quantile	mean	stdev	skewness	excess kurtosis
(0.99, 1]	0–5	ExpN	8.8931	82.9701	39.0396	16.9295	0.4758	3.0057
		Intensity	8.8984	82.2459	39.0582	16.8407	0.4416	2.9241
		NoF	8.7495	80.1854	38.7620	16.6059	0.4266	2.9748
		Poisson	8.8791	75.9685	37.2899	15.5257	0.3587	2.7286
		Constant	8.7407	82.1944	38.5745	16.8526	0.5340	3.3389
(0.99, 1]	6–10	ExpN	− 0.9155	63.3896	18.9734	12.3441	1.2197	4.9964
		Intensity	− 0.9805	63.3414	18.9901	12.3536	1.2208	5.0154
		NoF	− 1.0081	61.6145	19.0000	12.3482	1.1441	4.5849
		Poisson	− 1.0603	51.8066	17.7188	11.0212	1.0390	4.4388
		Constant	− 0.7940	62.5619	18.4307	11.8607	1.3183	5.4865
(1, 1.01]	0–5	ExpN	25.9912	88.6307	52.6331	12.8124	0.4068	3.6394
		Intensity	26.0735	86.3624	52.5518	12.6399	0.3532	3.5467
		NoF	25.9267	87.7606	52.2596	12.5129	0.3284	3.5106
		Poisson	23.7055	91.2434	51.2257	13.3583	0.4412	3.5094
		Constant	24.7890	86.8614	51.4311	12.5757	0.3723	3.5877
(1, 1.01]	6–10	ExpN	6.1377	71.8261	29.3628	12.9749	0.7704	3.9916
		Intensity	6.0673	71.5313	29.3201	12.9121	0.7474	3.9569
		NoF	5.6435	67.9710	29.2700	12.8706	0.6554	3.5968
		Poisson	5.1489	64.8335	28.3030	12.0505	0.5498	3.5151
		Constant	5.5137	71.4743	29.1097	12.8061	0.7211	3.9913
(1.01, 1.02]	0–5	ExpN	13.1870	83.6485	45.5686	15.8452	0.1484	3.1741
		Intensity	13.2705	82.8540	45.6146	15.8060	0.1386	3.0992
		NoF	12.9106	81.0215	45.0160	15.4066	0.0002	3.4576
		Poisson	14.3100	79.2772	44.0804	14.4855	0.2792	2.6911
		Constant	13.2680	81.3789	44.8105	15.5651	0.1067	3.3044
(1.01, 1.02]	6–10	ExpN	7.5791	60.8316	27.1647	12.2965	0.7350	3.0336
		Intensity	7.6802	62.2316	27.2055	12.3687	0.7763	3.1783
		NoF	7.4040	58.6456	27.0329	12.2208	0.6902	2.8973
		Poisson	6.6797	54.9548	26.1534	11.3747	0.6250	2.8537
		Constant	7.3923	64.8175	27.0419	12.6064	0.8579	3.4684

more obvious for the options that expire within 5 days. This is because trading on these options is usually more frequent than those ones with a longer maturity. The observed improvement is expected in general, because the market orders are self-exciting, so utilizing the past information can better predict the future arrivals of market orders and boost performance.

As for the comparison between ExpN and Intensity, neither one dominates the other for AAPL options, but ExpN outperforms by a small margin in all the cases for BAC options. The optimal quotes from these two strategies look similar in figures 6 and 7 with both exhibiting a U-shape for AAPL and an L-shape for BAC, which match the patterns in the marker orders (cf. Figure 2). However, the Intensity strategy can generate occasional large spikes in the quotes caused by surges in the Hawkes intensity. In contrast, the expected number of arrivals in the next period does not rise sharply with a surge in the intensity, so large spikes are not present in the ExpN strategy. We also notice that in many cases the performances of ExpN and Intensity are quite close to LIF, the unattainable upper bound that utilizes the exact number of market orders in the next time interval.

We compare the deep learning strategies with Constant, which keeps using the same quotes. The Poisson strategy, which is the deep learning strategy trained from Poisson arrivals, is inferior to Constant in every case, which is a simple strategy based on Hawkes arrivals. This result again demonstrates the importance of reaslitic modeling of market orders. ExpN and Intensity always outperform Constant, and their improvement can be considered quite significant given that the impractical upper bound LIF only outperforms Constant by about 4.3% $\sim$ 5.6% for the risk-neutral agent and about 0.9% $\sim$ 2.2% for the risk-averse agent.

The plots in figures 6 and 7 exhibit some interesting observations. For ExpN, Intensity and NoF, the optimal bid and ask quotes are always above \$0.1, which is the optimal distance to the mid-price if we know that there is only one market order arrival in the next time interval and maximize the expected profit of executing one limit order with $\kappa = 10$. As the expected number of market order arrivals in the next time interval is greater than one, the deep learning strategies post the limit order at a greater distance to the mid-price. In constrast, the Poisson strategy is learned from a less realistic dataset, which can produce quotes within the distance of \$0.1. In all these plots, we do not observe that the optimal bid or ask quote moves closer to the mid-price towards the end of the day. This is because there is no constraint on the option inventory in this setting. But our approach can generate this phenomenon when constraints are imposed (see figure 9(b)).

The daily PnL distributions of these strategies share similar characteristics for all the cases. They are all positively skewed with heavy tails. For each case, the magnitude of skewness and excess kurtosis of different strategies are quite close, but the tails could be quite different.

We conclude this section with two remarks. First, the differences among strategies shown in the paper are all out of sample. The in-sample differences are greater. Second, we

Table 10. Statistics of the daily PnL distribution for AAPL options with $\gamma = 0.05$.

K/S	TTM (days)	strategy	1% quantile	99% quantile	mean	stdev	skewness	excess kurtosis
(0.99, 1]	0–5	ExpN	8.8262	82.8967	38.8050	16.7088	0.4598	2.9809
		Intensity	8.8591	82.0616	38.7766	16.6221	0.4448	2.9469
		NoF	8.7764	78.0606	38.2569	16.0340	0.3374	2.7103
		Poisson	6.8725	92.7555	36.1628	16.8195	1.0453	5.7727
		Constant	8.8613	79.6093	38.3190	16.4984	0.4807	3.1023
(0.99, 1]	6–10	ExpN	− 0.8750	63.5448	19.0537	12.3961	1.2276	5.0033
		Intensity	− 0.9277	63.4896	19.0405	12.3797	1.2305	5.0131
		NoF	− 1.0073	60.6023	18.7006	12.0273	1.1386	4.6549
		Poisson	− 1.8365	58.6050	17.6924	12.0636	1.1726	4.5697
		Constant	− 0.9410	55.3669	18.5374	11.6998	1.1362	4.5949
(1, 1.01]	0–5	ExpN	26.0576	94.6963	52.8378	13.3135	0.5010	3.7588
		Intensity	26.0527	95.6309	52.8367	13.4275	0.5340	3.8416
		NoF	25.5957	84.7653	51.8810	12.1200	0.2883	3.5104
		Poisson	21.0371	106.0753	50.7553	16.6847	0.9310	4.5165
		Constant	24.7129	86.0199	51.3045	12.4110	0.3556	3.6519
(1, 1.01]	6–10	ExpN	6.2065	71.2012	29.3431	12.8534	0.7075	3.8336
		Intensity	6.2456	70.6717	29.2973	12.8118	0.7078	3.8181
		NoF	6.1406	68.4240	28.9356	12.4882	0.6346	3.6091
		Poisson	4.1470	69.5308	27.8440	12.7846	0.7119	4.0552
		Constant	5.7734	67.5115	28.8717	12.5281	0.6951	3.6991
(1.01, 1.02]	0–5	ExpN	13.2876	83.0615	45.5303	15.6362	0.2026	2.9114
		Intensity	13.2695	82.5931	45.5053	15.5702	0.1816	2.9175
		NoF	13.3809	79.4402	44.5230	14.6376	0.0892	3.0372
		Poisson	13.5962	79.0896	43.0733	14.6760	0.3617	2.8833
		Constant	13.5249	79.2305	44.1908	14.9214	0.1898	3.0098
(1.01, 1.02]	6–10	ExpN	7.8227	59.9707	27.1863	12.2043	0.7140	2.9264
		Intensity	7.8232	62.2686	27.2032	12.2877	0.7768	3.1835
		NoF	7.5886	60.3496	26.9074	11.9814	0.7210	3.0547
		Poisson	6.0749	59.6641	25.8119	11.8906	0.7866	3.2977
		Constant	7.5019	62.6192	26.5199	12.2838	0.8108	3.5488

Table 11. Statistics of the daily PnL distribution for BAC options.

K/S	TTM (days)	strategy	1% quantile	99% quantile	mean	stdev	skewness	excess kurtosis
				(a) $\gamma = 0$				
(0.98, 1]	0–5	ExpN	0.3994	14.1392	5.8036	3.2866	0.8018	4.2128
		Intensity	0.3979	14.1401	5.8021	3.2926	0.8099	4.2401
		NoF	0.3877	13.8154	5.7512	3.2326	0.6893	3.5946
		Poisson	0.4204	13.0855	5.5945	3.0111	0.5437	3.0329
		Constant	0.3584	14.2969	5.7664	3.3054	0.7365	3.7209
(1, 1.02]	0–5	ExpN	1.4517	17.2637	7.7385	2.8868	0.8855	4.5036
		Intensity	1.4502	17.2071	7.7290	2.8797	0.8655	4.4360
		NoF	1.4092	16.6128	7.6177	2.8285	0.7830	4.1479
		Poisson	1.4570	15.3550	7.5197	2.6199	0.6360	3.7694
		Constant	1.4346	16.6397	7.6190	2.8282	0.8020	4.1723
				(b) $\gamma = 0.05$				
(0.98, 1]	0–5	ExpN	0.4087	14.1187	5.8049	3.2847	0.7939	4.1604
		Intensity	0.3989	14.1226	5.8005	3.2923	0.8060	4.2244
		NoF	0.3916	13.7480	5.7497	3.2101	0.6540	3.4364
		Poisson	0.4214	12.8970	5.5693	2.9698	0.5157	2.9572
		Constant	0.3647	14.1318	5.7623	3.2641	0.6772	3.4509
(1, 1.02]	0–5	ExpN	1.4346	17.2930	7.7410	2.8911	0.8923	4.5329
		Intensity	1.4453	17.1490	7.7244	2.8727	0.8615	4.4213
		NoF	1.4131	16.5469	7.6274	2.8203	0.7667	4.1252
		Poisson	1.4609	15.2256	7.5109	2.5865	0.6114	3.7680
		Constant	1.4351	16.6397	7.6289	2.8131	0.7974	4.2006

have only considered trading one option contract at one or five minute frequency over one day. If we trade multiple contracts at a higher frequency over a longer period, the differences among the strategies can be more substantial.

REMARK 8 We did not consider trading costs in our study. For options, CBOE charges market makers trading fees, while many other options exchanges actually give rebates to market makers because they are liquidity providers. As our study

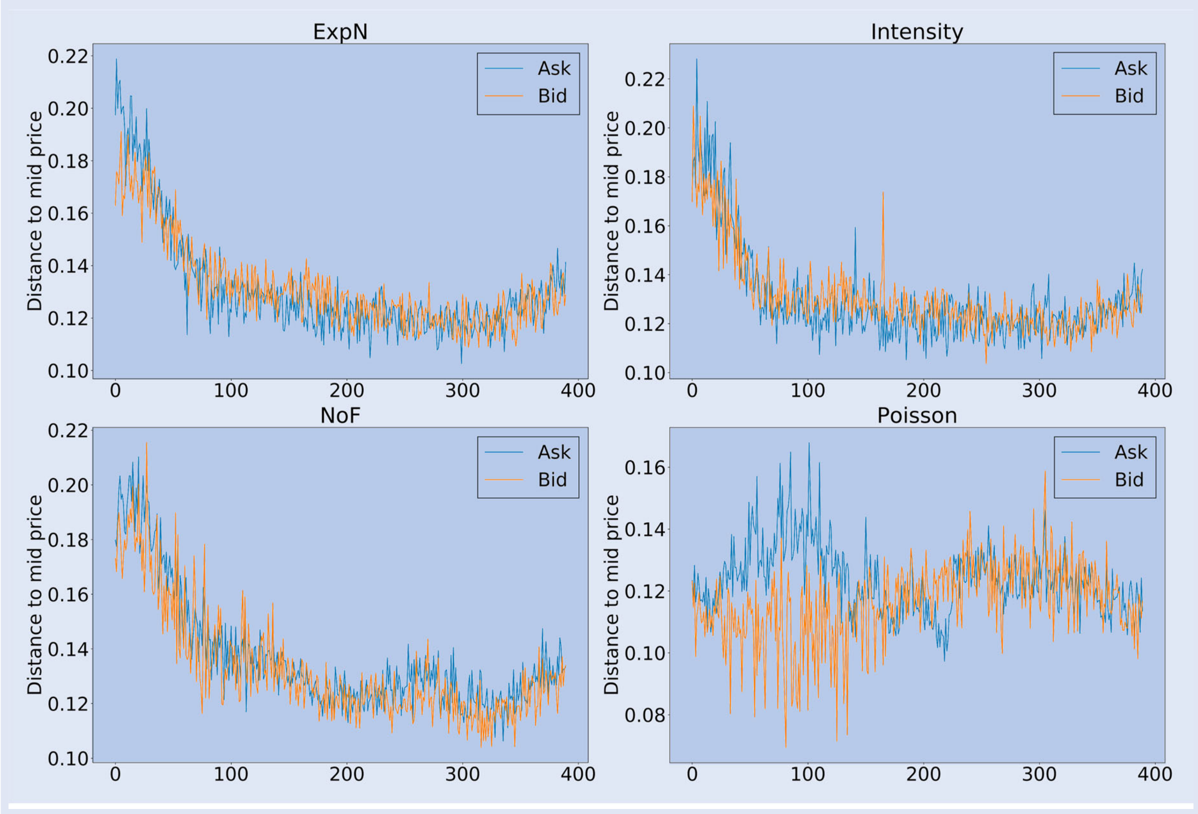

Figure 6. Strategies for one AAPL call option with $K/S \in (1.01, 1.02]$ and TTM = 8 days on 14 March 2019.

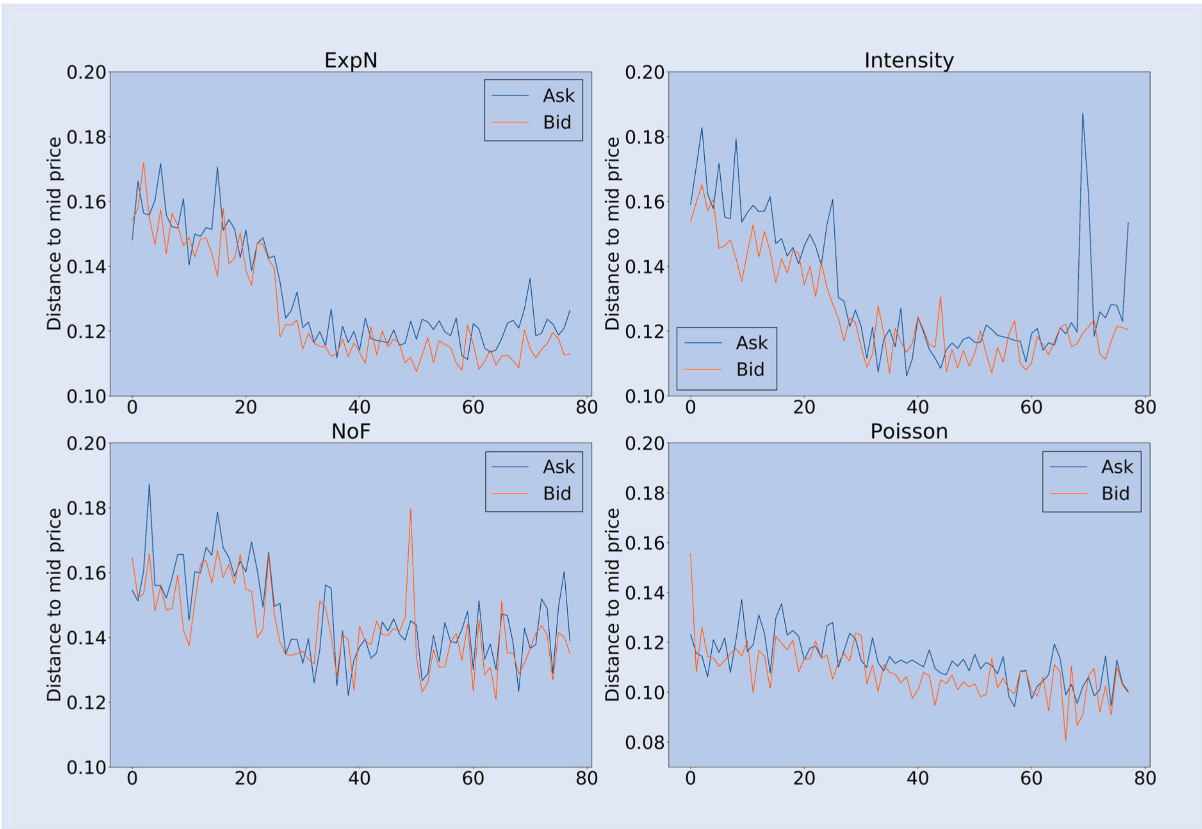

Figure 7. Strategies for one BAC call option with $K/S \in (1, 1.02]$ and TTM = 3 days on 19 March 2019.

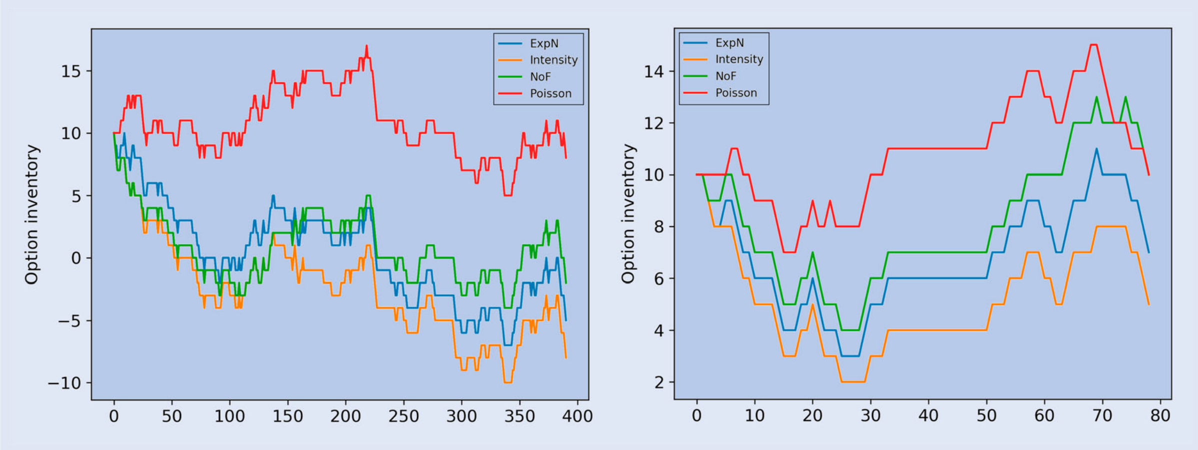

Figure 8. (a) Inventory of one AAPL call option with $K/S \in (1.01, 1.02]$ and TTM $= 8$ days on 14 March 2019. (b) Inventory of one BAC call option with $K/S \in (1, 1.02]$ and TTM $= 3$ days on 19 March 2019. (a) AAPL and (b) BAC.

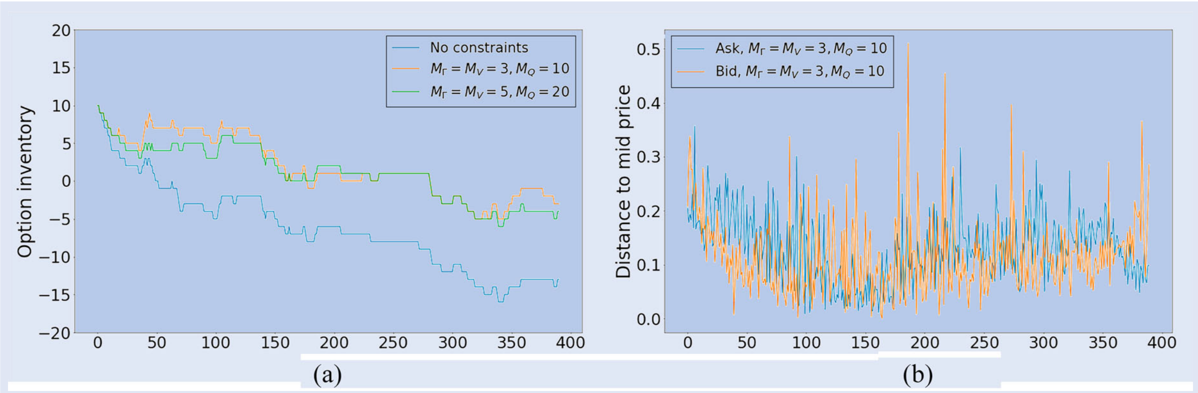

Figure 9. The inventory of one AAPL call option with $K/S \in (1, 1.01]$ and TTM $= 10$ days under three settings and the optimal strategy under Setting 3 on February 19, 2019. (a) Inventory and (b) Strategy.

does not specifically focus on market making on CBOE or any other options exchange, it is reasonable to report the clean performance of our market making strategies without considering trading costs and rebates. In our framework, it is easy to include them, as we can subtract the trading costs incurred from or add the rebates received to equation (9) that shows the change in cash. The remaining parts of our algorithm are not affected. Including trading costs would lower the market making profits for all strategies, but it is unlikely to change the main conclusions of our study that realistic modeling of market order flows matters and using the Hawkes features as inputs to the neural networks can boost performance.

5. Market making for multiple options with constraints

We extend the deep learning approach for option portfolios. Consider m options which are written on the same stock. The setting and objective of market making are the same as the single option case, but now constraints are imposed on the Greeks and inventory to limit the overnight risk, which is a significant source of risk for option dealers (Muravyev 2016). We assume the agent still applies delta hedging at each decision time. We add subscript l to the quantities to indicate that they are associated with the lth option. Equations (7)–(11) become

$$q^o_{k+1,l} = q^o_{k,l} + Q^b_{k,l}(\epsilon^b_{k,l}) - Q^a_{k,l}(\epsilon^a_{k,l}), \quad k = 0, \ldots, K-1, \quad l = 1, \ldots, m,$$

$$q^s_k = -\sum_{l=1}^m q^o_{k,l} \times \Delta_{k,l}, \quad k = 0, \ldots, K,$$

$$\begin{aligned} x_{k+1} - x_k = &\sum_{l=1}^m (C_{k,l} + \epsilon^a_{k,l}) \times Q^a_{k,l}(\epsilon^a_{k,l}) \\ &- (C_{k,l} - \epsilon^b_{k,l}) \times Q^b_{k,l}(\epsilon^b_{k,l}) \\ &- \Delta q^s_{k+1} S_{k+1}, \quad k = 0, \ldots, K-1, \end{aligned}$$

$$W_0 = x_0 + \sum_{l=1}^m q^o_{0,l} C_{0,l} + q^s_0 S_0,$$

$$W_K = x_K + \sum_{l=1}^m q^o_{K,l} C_{K,l} + q^s_K S_K.$$

5.1. The problem formulation

In this paper, we only consider gamma and vega for the Greeks. We require that the portfolio's gamma and vega as well as the inventory of each option are bounded, but for simplicity we only impose these bounds at the end of the time horizon. Since we only focus on a very short term (e.g.one day) for market making, this is quite reasonable. One can also impose these bounds at every time point in the horizon, which can be handled by the deep learning approach.

The market making problem is now formulated as

$$\begin{aligned} \max_{\substack{\epsilon^a_{k,l},\epsilon^b_{k,l}>0 \\ l=1,\dots,m \\ k=0,\dots,K-1}} \quad & \mathrm{E}[U(W_K - W_0)], \\ \text{s.t.} \quad & |\Gamma_K| \le M_\Gamma, \\ & |V_K| \le M_V, \\ & |q^o_{K,l}| \le M_{Q,l}, \quad \text{for } l = 1, \dots, m, \end{aligned}$$

where Γ_K and V_K refer to the gamma and vega of the option portfolio at t_K, and M_Γ, M_V, $M_{Q,l}$ are constants.

To solve the problem, we move the constraints to the objective function by adding them as penalties. We consider the quadratic penalty $P = \max\{0, x-y\}^2 + \min\{0, x+y\}^2$ for constraints like $|x| \le y$. The problem is now converted to

$$\begin{aligned} \min_{\substack{\epsilon^a_{k,l},\epsilon^b_{k,l}>0 \\ l=1,\dots,m \\ k=0,\dots,K-1}} \quad & -\mathrm{E}\Bigg[U(W_K - W_0) - \eta_1 P_g - \eta_2 P_v \\ & - \eta_3 \sum_{l=1}^{m} P_{q,l} \Bigg], \end{aligned}$$

where the penalty coefficients $\eta_1, \eta_2, \eta_3 > 0$, and P_g, P_v and $P_{q,l}$ are the quadratic penalties applied to the gamma, vega and inventory constraints. This unconstrained problem can then be solved by the deep learning algorithm in Section 3.3 with obvious modifications. Specifically, we now use $2m$ neural networks at each t_k, where there are two neural networks for each option to output its bid and ask quotes. By making the penalty coefficients large enough, we expect to recover the solutions of the original constrained problem.

5.2. Empirical results

We show the performance of the deep learning algorithm for the constrained problem using AAPL call options. We assume the agent is risk-neutral and again perform training and testing on a rolling basis as in the single option case. That is, for $n = 1, \dots, 11$, we use the data of month n to estimate the required parameters and test the strategy on the data of month $n+1$. We still consider market making at one minute frequency over one day. The option portfolio is formed by six options with one option from each of the six AAPL option groups. The market orders for each option are generated by the estimated Hawkes model for that option and we use the Hawkes intensity as an additional input to the neural networks. We set the initial cash $x_0 = 5000$, and the initial inventory of each option as 10. We consider three settings for comparison:

- Setting 1: no constraints;
- Setting 2: $M_\Gamma = M_V = 5$, $M_{Q,l} = M_Q = 20$ for all l;
- Setting 3: $M_\Gamma = M_V = 3, M_{Q,l} = M_Q = 10$ for all l.

Setting 3 is more stringent than Setting 2 by using smaller values for the upper bounds. To obtain appropriate values for the penalty coefficients, we increase their values until the violation rate on the validation set is small enough. Finally, we set $\eta_1 = \eta_2 = \eta_3 = 2000$ in our experiment.

Table 12 presents the out-of-sample results of these settings by showing the statistics of the daily PnL, $|\Gamma_K|$, $|V_K|$ and $|q^o_{K,l}|$ as well as the violation rates of the constraints. With a more stringent constraint, the average daily profit drops, which is expected. The standard deviation is the smallest in the most stringent case, which suggests the constraints can reduce the variability of the PnL. The violation rates of the three constraints in Settings 2 and 3 are all very small. In particular, there is no violation of the vega constraint. By using larger penalty coefficients for the gamma and inventory constraints, we can further drive their violation rates down to zero. We also observe significant reductions in the mean and standard deviation of $|\Gamma_K|$, $|V_K|$ and $|q^o_{K,l}|$ when they are constrained. This shows the usefulness of these constraints in limiting the overnight risk of holding the option portfolio.

Figure 9 plots the inventory of one AAPL option under three settings and the optimal strategy for the third setting on 19 February 2019. For the no constraint case, the inventory of the option goes down with time. While for the cases with constraints, the option inventory stays within the bounds throughout the day. As for the optimal ask quote, it moves closer to the mid-price towards the end of the day so that there is a higher chance of executing the limit sell order to reduce the option inventory to satisfy the terminal constraints.

Table 12. Test results of an option portfolio with AAPL call options.

		No constraints	$M_\Gamma = M_V = 5, M_Q = 20$	$M_\Gamma = M_V = 3, M_Q = 10$
PnL	mean	142.1131	140.0493	124.9877
	stdev	52.6806	53.6598	50.8760
$\lvert\Gamma_K\rvert$	Violation rate		0.49%	0.32%
	mean	1.3689	0.7100	0.3250
	stdev	1.4560	0.7181	0.3690
$\lvert V_K\rvert$	Violation rate		0	0
	mean	1.7915	0.6742	0.3442
	stdev	1.4894	0.5272	0.2886
$\lvert q^o_{K,l}\rvert$	Violation rate		0.09%	1.64%
	mean	9.5767	5.0845	2.8796
	stdev	7.3505	3.8202	2.5128

6. Conclusion

We develop a data-driven approach for options market making. Instead of imposing a parametric model, we start with an empirical analysis of the market order data for stock options. We document significant self-excitation in both buy and sell orders, but their cross-excitation is weak. We further show that a Hawkes process with a time-varying baseline intensity and the power law kernel provides a good fit to the market order data, which reveals its non-Markov nature. To tackle the market making problem based on non-Markov order arrivals, we develop a deep learning algorithm which can handle general model assumptions and constraints on the Greeks and inventory. In an out-of-sample study based on real market data, we demonstrate the importance of using realistic models and the Hawkes features for learning the market making strategy.

Acknowledgments

We thank the reviewers for their valuable comments that led to improvement in the paper.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This research was supported by Hong Kong Research Grant Council General Research Fund Grant 14206020.

References

Abergel, F., Huré, C. and Pham, H., Algorithmic trading in a microstructural limit order book model. *Quant. Finance*, 2020, **20**(8), 1263–1283.
Avellaneda, M. and Stoikov, S., High-frequency trading in a limit order book. *Quant. Finance*, 2008, **8**(3), 217–224.
Bacry, E. and Muzy, J.-F., Hawkes model for price and trades high-frequency dynamics. *Quant. Finance*, 2014, **14**(7), 1147–1166.
Bacry, E., Mastromatteo, I. and Muzy, J.-F., Hawkes processes in finance. *Mark. Microstruct. Liquid.*, 2015, **1**(1), 1550005.
Bacry, E., Jaisson, T. and Muzy, J.-F., Estimation of slowly decreasing hawkes kernels: Application to high-frequency order book dynamics. *Quant. Finance*, 2016, **16**(8), 1179–1201.
Baldacci, B., Derchu, J. and Manziuk, I., An approximate solution for options market-making in high dimension. arXiv preprint arXiv:2009.00907, 2020.
Baldacci, B., Bergault, P. and Guéant, O., Algorithmic market making for options. *Quant. Finance*, 2021, **21**(1), 85–97.
Bergault, P. and Guéant, O., Size matters for OTC market makers: General results and dimensionality reduction techniques. *Math. Finance*, 2021, **31**(1), 279–322.
Bowsher, C.G., Modelling security market events in continuous time: Intensity based, multivariate point process models. *J. Econom.*, 2007, **141**(2), 876–912.
Buehler, H., Gonon, L., Teichmann, J. and Wood, B., Deep hedging. *Quant. Finance*, 2019a, **19**(8), 1271–1291.
Buehler, H., Gonon, L., Teichmann, J., Wood, B., Mohan, B. and Kochems, J., Deep hedging: Hedging derivatives under generic market frictions using reinforcement learning. Avaiable from SSRN 3355706, 2019b.
Carbonneau, A. and Godin, F., Equal risk pricing of derivatives with deep hedging. *Quant. Finance*, 2021, **21**(4), 593–608.
Cartea, Á. and Jaimungal, S., Risk metrics and fine tuning of high-frequency trading strategies. *Math. Finance*, 2015, **25**(3), 576–611.
Cartea, A. and Jaimungal, S., Incorporating order-flow into optimal execution. *Math. Financ. Econ.*, 2016, **10**(3), 339–364.
Cartea, Á., Jaimungal, S. and Ricci, J., Buy low, sell high: A high frequency trading perspective. *SIAM J. Financ. Math.*, 2014, **5**(1), 415–444.
Cartea, Á., Donnelly, R. and Jaimungal, S., Algorithmic trading with model uncertainty. *SIAM J. Financ. Math.*, 2017, **8**(1), 635–671.
Cartea, A., Donnelly, R. and Jaimungal, S., Enhancing trading strategies with order book signals. *Appl. Math. Finance*, 2018a, **25**(1), 1–35.
Cartea, A., Jaimungal, S. and Ricci, J., Algorithmic trading, stochastic control, and mutually exciting processes. *SIAM Rev.*, 2018b, **60**(3), 673–703.
Choi, S.E., Jang, H.J., Lee, K. and Zheng, H., Optimal market-making strategies under synchronised order arrivals with deep neural networks. *J. Econ. Dyn. Control*, 2021, **125**, 104098.
Confortola, F. and Fuhrman, M., Backward stochastic differential equations and optimal control of marked point processes. *SIAM J. Control Optim.*, 2013, **51**(5), 3592–3623.
Dai, Z., Li, L. and Zhang, G., Evaluation of deep learning algorithms for quadratic hedging. *J. Deriv.*, 2022, **30**(1), 32–57.
El Aoud, S. and Abergel, F., A stochastic control approach to option market making. *Mark. Microstruct. Liquid.*, 2015, **1**(1), 1550006.
Fecamp, S., Mikael, J. and Warin, X., Deep learning for discrete-time hedging in incomplete markets. *J. Comput. Finance*, 2020, **25**(2), 51–85.
Fodra, P. and Pham, H., High frequency trading and asymptotics for small risk aversion in a Markov renewal model. *SIAM J. Financ. Math.*, 2015, **6**(1), 656–684.
Gao, X., Zhou, X. and Zhu, L., Transform analysis for Hawkes processes with applications in dark pool trading. *Quant. Finance*, 2018, **18**(2), 265–282.
Gašperov, B. and Kostanjčar, Z., Market making with signals through deep reinforcement learning. *IEEE. Access.*, 2021, **9**, 61611–61622.
Gašperov, B. and Kostanjčar, Z., Deep reinforcement learning for market making under a hawkes process-based limit order book model. *IEEE Control Syst. Lett.*, 2022, **6**, 2485–2490.
Gašperov, B., Begušić, S., Posedel Šimović, P. and Kostanjčar, Z., Reinforcement learning approaches to optimal market making. *Mathematics*, 2021, **9**(21), 2689.
Guéant, O. and Manziuk, I., Deep reinforcement learning for market making in corporate bonds: Beating the curse of dimensionality. *Appl. Math. Finance*, 2019, **26**(5), 387–452.
Guéant, O., Lehalle, C.-A. and Fernandez-Tapia, J., Dealing with the inventory risk: A solution to the market making problem. *Math. Financ. Econ.*, 2013, **7**(4), 477–507.
Guilbaud, F. and Pham, H., Optimal high-frequency trading with limit and market orders. *Quant. Finance*, 2013, **13**(1), 79–94.
Guilbaud, F. and Pham, H., Optimal high-frequency trading in a pro rata microstructure with predictive information. *Math. Finance*, 2015, **25**(3), 545–575.
Hambly, B., Xu, R. and Yang, H., Recent advances in reinforcement learning in finance. arXiv preprint arXiv:2112.04553, 2021.
Han, J. and E, W., Deep learning approximation for stochastic control problems. In *Advances in Neural Information Processing Systems*, Deep Reinforcement Learning Workshop, 2016.
Ho, T. and Stoll, H.R., Optimal dealer pricing under transactions and return uncertainty. *J. Financ. Econ.*, 1981, **9**(1), 47–73.
Kingma, D.P. and Ba, J., Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.

Kumar, P., Deep Hawkes process for high-frequency market making. arXiv preprint arXiv:2109.15110, 2021.
Lai, Q., A data-driven approach for option market making. Ph.D. Thesis, The Chinese University of Hong Kong, 2021.
Law, B. and Viens, F., Market making under a weakly consistent limit order book model. *High Freq.*, 2019, **2**(3), 215–238.
Lee, C.M. and Ready, M.J., Inferring trade direction from intraday data. *J. Finance.*, 1991, **46**(2), 733–746.
Muravyev, D., Order flow and expected option returns. *J. Finance.*, 2016, **71**(2), 673–708.
Potters, M. and Bouchaud, J.P., More statistical properties of order books and price impact. *Phys. A: Stat. Mech. Appl.*, 2002, **324**(1), 133–140.
Reppen, A.M. and Soner, H.M., Deep empirical risk minimization in finance: Looking into the future. *Math. Finance*, 2022, **33**(1), 116–145.
Savickas, R. and Wilson, A.J., On inferring the direction of option trades. *J. Financ. Quant. Anal.*, 2003, **38**(4), 881–902.
Stoikov, S. and Sağlam, M., Option market making under inventory risk. *Rev. Deriv. Res.*, 2009, **12**(1), 55–79.
Zhang, J. and Huang, W., Option hedging using LSTM-RNN: An empirical analysis. *Quant. Finance*, 2021, **21**(10), 1753–1772.
Zhao, M. and Linetsky, V., High frequency automated market making algorithms with adverse selection risk control via reinforcement learning. In *Proceedings of the Second ACM International Conference on AI in Finance*, pp. 1–9, 2021.