ECONOMIC RESEARCH
FEDERAL RESERVE BANK OF ST. LOUIS

WORKING PAPER SERIES

Sluggish news reactions: A combinatorial approach for synchronizing stock jumps

Authors	Nabil Bouamara, Kris Boudt, Sébastien Laurent, and Christopher J. Neely
Working Paper Number	2024-006B
Revision Date	October 2025
Citable Link	https://doi.org/10.20955/wp.2024.006
Suggested Citation	Bouamara, N., Boudt, K., Laurent, S., Neely, C.J., 2025; Sluggish news reactions: A combinatorial approach for synchronizing stock jumps, Federal Reserve Bank of St. Louis Working Paper 2024-006. URL https://doi.org/10.20955/wp.2024.006

Federal Reserve Bank of St. Louis, Research Division, P.O. Box 442, St. Louis, MO 63166

Sluggish news reactions: A combinatorial approach for synchronizing stock jumps[*]

Nabil Bouamara
Data Science Center, National Bank of Belgium

Kris Boudt
Department of Economics, Ghent University
Solvay Business School, Vrije Universiteit Brussel
School of Business and Economics, Vrije Universiteit Amsterdam

Sébastien Laurent
Aix Marseille Univ., CNRS, AMSE, Marseille, France
Aix-Marseille Graduate School of Management-IAE

Christopher J. Neely
Research Division, Federal Reserve Bank of St. Louis

October 30, 2025

Abstract

Stock prices often react sluggishly to news, producing gradual and delayed jumps. Econometricians typically treat these sluggish reactions as microstructure effects and settle for a coarse sampling grid to guard against them. We introduce new methods to synchronize mistimed stock returns on a fine sampling grid that allow us to better approximate the true common jumps in the efficient prices of related stocks in an application to Dow 30 data. The synchronized jumps produce better jump covariance estimates and estimates of the realized jump betas with better forecasting power, and superior trading rule performance.

Keywords: Asynchronicity; Cojumps; High-frequency data; Microstructure noise; Realized Covariance; Rearrangement

JEL: C02, C58, G11, G14

[*] We thank Anna Cole for excellent research assistance. We have received helpful comments and suggestions from Geert Dhaene, Jean-Yves Gnabo, Roxana Halbleib, Ilze Kalnina, Nathan Lassance, Oliver Linton, André Lucas, Kristien Smedts, Steven Vanduffel, and the conference and seminar participants at KU Leuven, Vrije Universiteit Brussel, Vrije Universiteit Amsterdam, the Computational and Financial Econometrics Conference (2021), the Belgian Financial Research Forum (2023), the Quantitative and Financial Econometrics Conference (2023), and the Society of Financial Econometrics Summer School (2023). Nabil Bouamara gratefully acknowledges support from the Flemish Research Foundation (FWO fellowship #11F8419N) and the Platform for Education and Talent (Gustave Boël – Sofina fellowship). Sébastien Laurent has received fundings from the French government under the "France 2030" investment plan managed by the French National Research Agency (reference: ANR-17-EURE-0020 and ANR-21-CE26-0007-01) and from Excellence Initiative of Aix-Marseille University - A*MIDEX. The views expressed in this paper are those of the author(s) and do not necessarily reflect the views of the Federal Reserve System, the Board of Governors, the regional Federal Reserve Banks, or the National Bank of Belgium.

1 Introduction

Financial economics predicts that asset prices adjust rapidly to new information. Major economic news, such as FOMC announcements, fiscal news, natural disasters, and geopolitical conflicts, can produce synchronized large price changes — so-called cojumps — across multiple assets. A systematic cojump is a common discontinuity among stock prices that produces a jump in the price of the market portfolio. Such discontinuities have important implications for portfolio selection and hedging because investors cannot diversify them away (Gilder et al., 2014), and the desire to keep jump-risk exposure below a certain level dominates an investor's portfolio strategy (Aït-Sahalia et al., 2009).

Empirical finance seeks to estimate the volatility, dependence and extreme changes in the prices of financial assets. The use of very high-frequency data allows these quantities to be estimated precisely, but poses at least two major problems: 1) the Epps effect[1] biases estimates of covariances/correlations in asset returns, and 2) microstructure noise can bias estimates of realized volatility calculated with very high-frequency data.

In practice, observed prices do not always immediately change with the underlying efficient price, but rather react "sluggishly". Bandi et al. (2017) find that small price adjustments are often smaller than one would expect from the underlying semimartingale assumption.[2] A different form of sluggishness can arise for price jumps. Li, Todorov, Tauchen, and Chen (2017) find that jumps of less-liquid individual assets typically lag those of the more-liquid market index, while Barndorff-Nielsen et al. (2009) report that jumps may develop gradually, with a sequence of smaller jumps instead of a single large jump. Sluggish alignment of transaction prices with their efficient counterparts produces asynchronous observed jumps that lead to an underestimation of cojump frequency and asset price dependence. In other words, mistimed jumps produce an effect analogous to the Epps (1979) effect. This mistiming is a problem because standard cojump tests implicitly assume that transaction prices across assets adjust simultaneously (e.g., Bollerslev et al., 2008; Bandi and Renò, 2016). Most researchers have dealt with this problem by settling for a coarse sampling grid, e.g., 5- or 10-minute data (Barndorff-Nielsen et al., 2009; Bollerslev et al., 2008; Jacod et al., 2009; Lahaye et al., 2011; Li et al., 2019), but while a coarse grid reduces microstructure effects, it smooths price changes too much (Aït-Sahalia, 2004).

We propose an alternative method of sampling data to recover underlying efficient prices and the common jumps in a basket of stocks. Our method is not a new jump test but rather a novel method of organizing data, in the way that refresh-time sampling and time aggregation (i.e., coarser sampling) are methods of organizing data. Our method assumes that the carefully watched exchange-traded fund (ETF) price tracks the latent, equilibrium (also often called "efficient") value of a stock index, but that sluggish news impoundment to the price of individual stocks can create a temporary spread between the value of a synthetic index and the efficient ETF price.

To minimize this index-ETF spread, we extend and apply the combinatorial methods of Puccetti and Rüschendorf (2012) and Embrechts et al. (2013) to optimally rearrange

[1] Epps (1979) shows the estimated correlation between two assets returns becomes more biased toward zero as the sampling interval gets shorter, e.g., from 1-minute to 1-second.

[2] Bandi et al. (2017) define sluggish prices as price adjustments smaller (in absolute value) than a given threshold. They show that the fraction of sluggish prices converges to zero if the price process is a semimartingale, but tends to a number between 0 (not included) and 1 if the price dynamics are staler than implied by the ubiquitous semimartingale assumption.

asynchronous jumps in time. This algorithm of Puccetti and Rüschendorf (2012) and Embrechts et al. (2013) is best known for actuarial applications, but has also been applied to other disciplines, such as option pricing (e.g., Bondarenko and Bernard, 2024) and operations research (e.g., Boudt et al., 2018). We denote our extension of these combinatorial methods as the "Rearrangement Mixed-Integer Linear Program". Rearrangement improves covariance estimation without sacrificing high-frequency precision.

We illustrate the method using New York Stock Exchange data for the Dow 30 stocks and the DIA ETF. Rearranged jumps tend to coincide with major macroeconomic announcements – most notably those related to Federal Reserve communications – supporting the interpretation of sluggish price adjustment to systemic news. The results show that the proposed rearrangement successfully recovers common, efficient jumps that are otherwise scattered across individual stocks. The rearranged returns improve the estimation of realized jump covariances, enhance the predictive power of realized jump betas in forecasting daily returns and improve trading rules.

We proceed as follows. Section 2 details the synchronization method using a toy example. Section 3 illustrates an empirical example of asynchronous reactions to news in the Dow 30 and shows the impact of the rearrangement on the estimation of realized jump betas. Section 4 concludes. Additional implementation details and discussions are provided in the Online Supplement.

2 Synchronizing jumps: A combinatorial problem

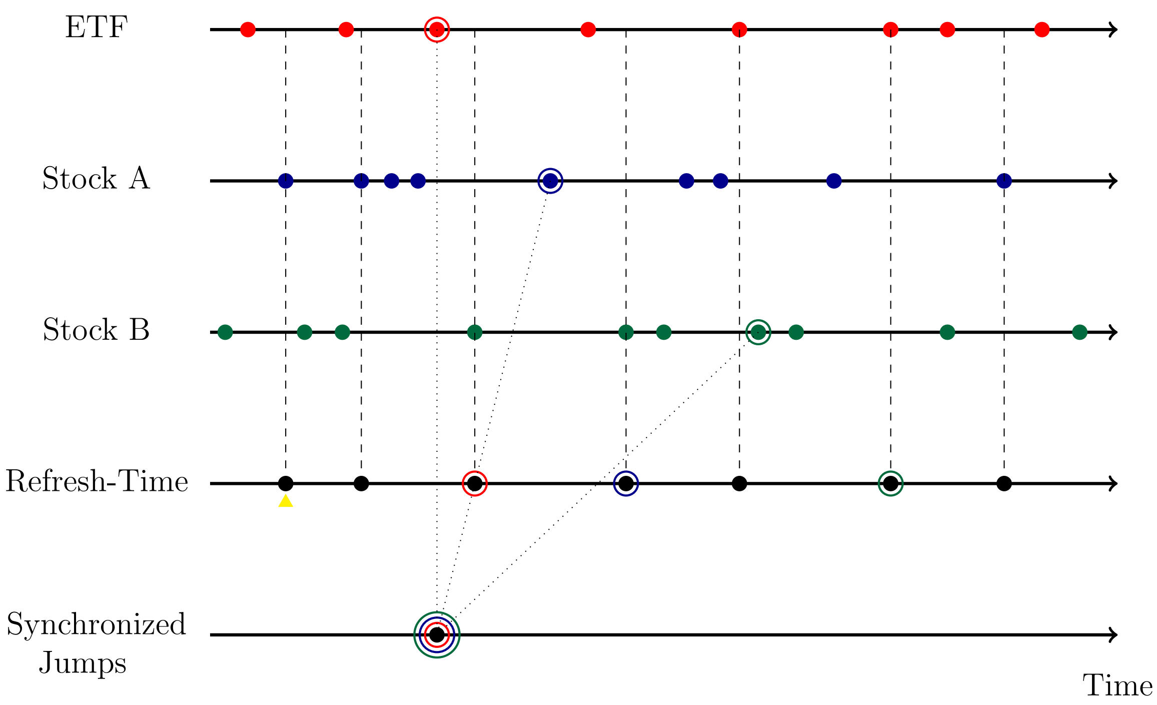
Non-synchronous trading at high frequencies leads to so-called stale prices. Addressing asynchronicity through coordinated data collection and organization has been integral to covariance estimations since at least Epps (1979) and remains an active area of financial research, see *e.g.*, Barndorff-Nielsen et al. (2011) or Boudt et al. (2017) and the references therein. Despite this progress, existing sampling schemes like refresh-time sampling (Barndorff-Nielsen et al., 2011) are not designed to handle asynchronous reactions to news, because stale prices are not the only cause of asynchronous jumps. Sometimes asset prices may react "sluggishly" to the same news event – that is, news may be impounded into their prices with a delay, even while trading continues. To address this problem, we coordinate the timing of multivariate jumps (news reactions) using what we call 'Jump Synchronization'. This method of data organization realigns mistimed jumps across assets, so that delayed reactions to a common news event are synchronized.

Figure 1 compares refresh-time sampling to jump synchronization in the presence of asynchronous observations and jumps. It draws inspiration from Figure 1 in Barndorff-Nielsen et al. (2011), which illustrates refresh-time sampling in a situation with three assets (in the absence of jumps). We expand upon this concept to include scenarios with asynchronous jumps in prices of three assets: a basket instrument and its two underlying stocks. For each asset, the filled dots in Figure 1 indicate the updates in posted prices, and an open dot represents price jumps. Vertical dashed lines represent the sampling times generated from the three assets using refresh-time sampling. For example, the first black dot (highlighted by the yellow triangle) represents the time it takes for all three assets to trade. Because (il)liquidity issues[3] are not the sole cause(s) of asynchronous jumps,

[3] Asynchronous jumps in asset prices happen for several reasons, not just differences in liquidity. Assets may react differently to news based on the traders involved, their beliefs, and actions. For example, major news like the FOMC statement on September 18, 2007, discussed in Section 3.3, triggered an increase

however, refresh-time sampling alone does not resolve the problem of asynchronous jumps.

Figure 1: Jump synchronization



Note: This diagrams compares refresh-time sampling to jump synchronization in the presence of asynchronous observations and jumps. In each asset's case, the filled dots indicate when the posted prices were updated, and the open dot represents the time at which the price jumps. The vertical dashed lines represent the sampling times generated from the three assets using refresh-time sampling. The first black dot, highlighted by the yellow triangle, marks the first refresh-time, i.e., the moment when all three assets have traded at least once. Jump synchronization rearranges mistimed jumps to occur simultaneously with the ETF jump.

To confront asynchronicity in price adjustments, jump synchronization rearranges mistimed jumps so that they occur simultaneously with the ETF jump. The lower section of the diagram illustrates how jumps (open dots) in Stocks A and B are realigned with the earlier jump in the ETF. The next subsection details how we optimally rearrange jumps while prohibiting economically implausible rearrangements. We use a simulated example to clarify the mechanics of the rearrangements.

2.1 A DGP for sluggish news reactions

We assume a data generating process with gradual and delayed stock price jumps.[4] Trading frictions may prevent the observed prices from immediately reflecting a jump in the

in trading, unlike the typical activity seen throughout the rest of the day (not reported). This suggests that time variation in trading intensity contributes to asynchronous jumps.

[4] Gradual jumps are when the prices exhibit strong linear trends for periods of a few minutes (Barndorff-Nielsen et al., 2009). Jump delays are when jumps of individual assets follow those of the highly liquid market index during market-wide events (Li, Todorov, Tauchen, and Chen, 2017).

underlying equilibrium price. The conventional martingale-plus-noise price model does not capture such important complications for multivariate jump analysis.

Let $X_t = (X_{1,t}, ..., X_{p,t})^\top$ denote the logarithmic p-variate, equilibrium (or so-called "efficient") price of the p stocks in the market index. The price process is defined on a filtered probability space $(\Omega, \mathcal{F}, (\mathcal{F})_{t\geq 0}, \mathbb{P})$ and is adapted to the filtration $\mathcal{F}_t$ that represents information available to market participants at time $t \geq 0$. We assume that X operates in an arbitrage-free, frictionless market, which implies that X is a semi-martingale. Econometricians (see Aït-Sahalia and Jacod, 2014) model stock prices as a jump-diffusion process:

$$\begin{aligned} X_t &= X_t^c + X_t^d, \text{ with,} \\ X_t^c &:= X_0 + \int_0^t b_s ds + \int_0^t \sigma_s dW_s, \\ X_t^d &:= \sum_{s\leq t} \Delta X_s, \end{aligned} \tag{1}$$

in which X_t^c (resp. X_t^d) is the continuous (resp. discontinuous) part of X_t, b is the drift, σ is the stochastic (co)volatility, W is a multivariate Brownian motion, and $\Delta X_t := X_t - X_{t-}$, where X_{t-}, the left limit at time t, denotes the jumps of X at time t. A stock's growth prospects generate a jump in single stock price (Christensen et al., 2023; Lee and Mykland, 2008) but major economic news, such as pre-scheduled announcements, natural disasters or geopolitical conflicts, trigger common (*i.e.*, synchronous) jumps in related stock prices (see Li, Todorov, Tauchen, and Chen, 2017, for examples). Several cojumps tests have been proposed in the literature, including Jacod et al. (2009), Caporin et al. (2017), Bandi and Renò (2016), Gnabo et al. (2014), and Bibinger and Winkelmann (2015).

In practice, we observe discretely sampled, noisy transaction prices, rather than the latent process in (1). Frictions such as tick size, discrete observations, bid-ask spreads, adverse selection, liquidity, and inventory control produce market microstructure noise (see *e.g.*, Christensen et al., 2014; Lee and Mykland, 2012).

To account for this, we rewrite the logarithmic price process $Y_t = (Y_{1,t}, ..., Y_{p,t})^\top$ for p stocks as the sum of two components: $Y_{i\Delta_n}^c$, a noisy observation of the continuous price component $X_{i\Delta_n}^c$, and $Y_{i\Delta_n}^d$, a noisy observation of the discontinuous price component ΔX_s^d:

$$\begin{aligned} Y_{i\Delta_n} &= Y_{i\Delta_n}^c + Y_{i\Delta_n}^d, \text{ with} \\ Y_{i\Delta_n}^c &:= X_{i\Delta_n}^c + u_{i\Delta_n}, \\ Y_{i\Delta_n}^d &:= \sum_{s\leq i\Delta_n} b_{s,i\Delta_n} \Delta X_s. \end{aligned} \tag{2}$$

Here, $i = 0, ..., \lfloor T/\Delta_n \rfloor$ indexes the regular observation times across a fixed time span T (e.g., one trading day), Δ_n denotes the width of the sampling interval, and $\lfloor \cdot \rfloor$ is the floor function. The term $u_{i\Delta_n}$ represents classical market microstructure noise, while the jump component of prices is modeled as a weighted sum of past and contemporaneous efficient price jumps. The weights $b_{s,i\Delta_n}$ determine the progress to efficiency, that is, the fraction of an efficient price jump at time s that has been incorporated into observed prices by time $i\Delta_n$. Typically, $b_{s,i\Delta_n} = 0$ before the jump in the efficient price ($i\Delta_n < s$) and $b_{s,i\Delta_n} = 1$ once prices have fully adjusted ($i\Delta_n \geq s$). In the presence of sluggish adjustment or overreaction, however, $b_{s,i\Delta_n}$ evolves as a step function that gradually reaches (or temporarily exceeds) 1, capturing the delayed (or overshooting) impoundment of information into prices.

There are thus two types of noise in (2): one affecting the continuous part and another affecting the discontinuous part. Microstructure noise u contaminates the efficient price process X, but is typically too small to substantially contaminate the discontinuous part X^d. By itself, it cannot generate gradual jumps (as in Barndorff-Nielsen, Hansen, Lunde, and Shephard, 2009) or delayed jumps (as in Li et al., 2017). We instead capture mistimed or mismeasured jumps in a separate noisy jump component Y^d, which allows for sluggish news reactions that spread a single efficient jump over several time intervals. We refer to these as “sluggish jumps”, referring to delayed or gradual adjustments of observed prices to an underlying efficient price jump.

Figure 2 shows a simulated sample path of this new DGP (2) for second-by-second prices of a single stock (see Appendix A of the Online Supplement for details on the simulation setup). The gray line in the top panel of Figure 2 illustrates that the efficient stock price jumps at 12:45 in reaction to news. In the following 112 seconds, the observed price (in black) catches up with the new equilibrium level (in gray) by gradually matching the jump. The middle and bottom panels, respectively, decompose the price process into its continuous component and its discontinuous component. The middle panel compares the efficient continuous price trajectory with that contaminated by market microstructure noise. The bottom panel compares the efficient, abrupt jump with the contaminated, more gradual jump. Appendix A of the Online Supplement provides further implementation details on how to spread the jump over several time intervals.

A coarser sampling grid, *i.e.*, at a lower sampling frequency Δ_n, reduces the impact of noise but oversmooths changes (as in Bollerslev et al., 2008; Lahaye et al., 2011; Li et al., 2019), which reduces the probability that one can recognize a jump (Aït-Sahalia, 2004).

2.2 Collecting asynchronous jumps in a jump-event matrix

When multiple stocks react sluggishly to new information, their jumps will generally be asynchronous on a fine sampling grid, and these jumps will not all coincide with the jump in the price of an index-tracking ETF. Empirically, jumps of less-liquid individual assets typically lag those of the more-liquid market index, (Li, Todorov, Tauchen, and Chen, 2017) and the ETF price jumps more often than that of a synthetic stock index price (Bollerslev et al., 2008).

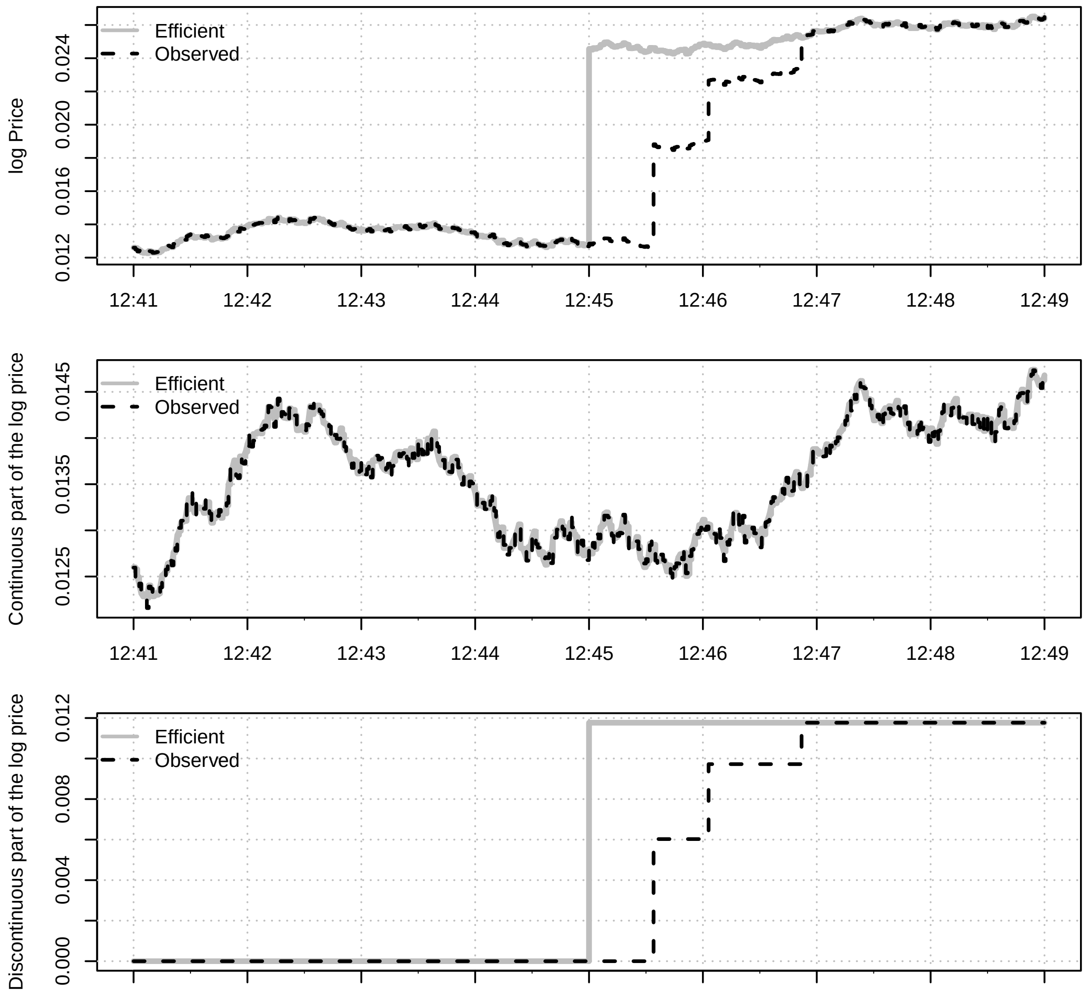
2.2.1 The spread measures sluggishness in high-frequency data

Let $w_{k,t}$, with $k = 1, \ldots, p$, be the weights allocated to each stock in the market index at each moment in time. The price of the synthetically constructed index portfolio, $S_{i\Delta_n}$, is a linear combination of the observed stock prices in (2), sluggishly incorporating jumps:

$$S_{i\Delta_n} = \sum_{k=1}^{p} w_{k,i\Delta_n} Y_{k,i\Delta_n}. \tag{3}$$

We assume that the observed log price of an ETF, Z_t, efficiently prices the index of

Figure 2: The discontinuous component of a stock's price may react sluggishly to news



Note: We plot the decomposition of the efficient (1) and observed (2) log price process for one simulated, sample path during a short event window. The top panel shows the log price. The middle panel shows the continuous part of the log price, and the bottom panel shows the discontinuous part of the log price. The efficient stock price jumps at $i\Delta_n = 11,701$ or 12:45. Market microstructure noise contaminates the continuous part of the price process, while delays in the jump process contaminate the discontinuous part of the price process.

the p stocks (1), incorporating jumps in the p efficient stock prices:[5]

$$Z_t = \sum_{k=1}^{p} w_{k,t} X_{k,t}. \tag{4}$$

The deviation or spread in prices, denoted $\delta^p_{i\Delta_n}$, is the difference between the observed price of a synthetic stock index (3) and the observed price of an ETF tracking the index (4):

$$\delta^p_{i\Delta_n} := S_{i\Delta_n} - Z_{i\Delta_n}. \tag{5}$$

[5] To simplify our notation, we rely on a weighted average of individual log returns rather than simple returns. This difference is minor in empirical applications (Jondeau, Poon, and Rockinger, 2007, pp. 9).

We can define the return spread as the difference between the observed returns on a synthetic stock index, $\Delta_i^n S := \sum_{k=1}^p w_{k,i\Delta_n} \Delta_i^n Y_k$, and the observed returns of an ETF on the index, $\Delta_i^n Z := Z_{i\Delta_n} - Z_{(i-1)\Delta_n}$, or, equivalently, the percentage change of the price spread in (5):

$$\delta^r_{i\Delta_n} := \delta^p_{i\Delta_n} - \delta^p_{(i-1)\Delta_n} = \Delta_i^n S - \Delta_i^n Z. \tag{6}$$

In the absence of microstructure noise and/or sluggish jumps, we would expect the ETF log price Z to equal the synthetic log price S. However, these phenomena cause the price of the synthetic index portfolio to deviate from the presumably efficient price of an ETF tracking the index.[6] The sluggish jump component is much larger than microstructure noise, so asynchronous impoundment of news drives the spread in prices. Hence, the spread (5) between the ETF price and a synthetically constructed index measures "sluggishness" – the collective misalignment of noisy stock prices with their efficient levels. We seek to rearrange jumps in time to minimize the spread and approximately recover the latent efficient price.

A stylized 3-stock universe ($p = 3$) and a corresponding ETF illustrate our procedure. The stock names A, B and C correspond to the indices $k = 1, 2$ and 3. The ABC ETF price is an equally weighted average of the underlying stocks' efficient prices (4), and the synthetic ABC portfolio price is an equally weighted average of observed stock prices (3). The sampling frequency is one minute, i.e., $\Delta_n = 1/390$.

Each row in the vectors below represents a percentage return of the three stocks and the ABC ETF in one time interval. The jump returns are underlined. For example, in the first minute, stock A's price fell 0.018 percent, stock B's price rose 0.015 percent, stock C's price fell 0.12 percent, and the ABC ETF price fell 0.039 percent.

$$\Delta_i^n Y_1 = \begin{bmatrix} \vdots \\ -0.018 \\ -0.031 \\ -0.057 \\ \underline{0.629} \\ \underline{0.651} \\ \vdots \end{bmatrix}, \Delta_i^n Y_2 = \begin{bmatrix} \vdots \\ 0.015 \\ -0.067 \\ -0.029 \\ \underline{1.201} \\ 0.062 \\ \vdots \end{bmatrix}, \Delta_i^n Y_3 = \begin{bmatrix} \vdots \\ -0.120 \\ -0.104 \\ 0.088 \\ 0.017 \\ 0.074 \\ \vdots \end{bmatrix}, \text{and } \Delta_i^n Z = \begin{bmatrix} \vdots \\ -0.039 \\ -0.071 \\ \underline{0.807} \\ 0.001 \\ 0.073 \\ \vdots \end{bmatrix}. \tag{8}$$

The timings of these jumps shows that stock prices asynchronously incorporate news. The ETF's price (last column) jumps 0.807 percent in the third minute. Stock A's price (first column) jumps late and gradually in 2 segments and takes 2 minutes longer than the ETF's price (last column) to complete its jumps. Stock B's jump (second column) is delayed by one minute. That is, the information in the ETF jump is not impounded into stock B's price for a minute. Stock C does not jump at all (third column).

[6] Within our theoretical model descriptions, the difference between the price of the synthetic index and the ETF price (5) equals a microstructure noise component minus the discontinuous component that has not yet been impounded into the observed prices:

$$S_{i\Delta_n} - Z_{i\Delta_n} = \sum_{k=1}^p w_{k,i\Delta_n} u_{k,i\Delta_n} + \sum_{k=1}^p w_{k,i\Delta_n} \left(Y^d_{k,i\Delta_n} - X^d_{k,i\Delta_n} \right). \tag{7}$$

This sharp theoretical decomposition is unobservable to the econometrician in empirical data.

These delays cause the implied (inefficient) returns of the synthetic ABC portfolio to deviate from the (efficient) returns of the ABC ETF. The sum of the first three columns of the matrix below (equation (9)) is the return to the synthetic ABC portfolio, that is, the equally weighted sum of the log returns of stocks A, B, and C (the first three columns in Equation (8)). The fourth column on the left side of the equal sign in Equation (9) is the efficient log return of the ABC ETF (the last column in (8)). The difference between the return to the synthetic ABC portfolio and the ABC ETF is on the right-hand side of Equation (9). This difference is the return spread, i.e., $\delta^r_{i\Delta_n}$ in (6), in each of the five periods.

$$\delta^r_{i\Delta_n} = \left[\begin{array}{ll} & \vdots \\ {}^1\!/_3(-0.018+0.015-0.120) & -(-0.039) \\ {}^1\!/_3(-0.031-0.067-0.104) & -(-0.071) \\ {}^1\!/_3(-0.057-0.029+0.088) & -\quad \underline{0.807} \\ {}^1\!/_3(\ \ \underline{0.629}+\underline{1.201}+0.017) & -\quad 0.001 \\ {}^1\!/_3(\ \ \underline{0.651}+0.062+0.074) & -\quad 0.073 \\ & \vdots \\ \underbrace{\hphantom{{}^1\!/_3(-0.018+0.015-0.120)}}_{\substack{\text{ABC portfolio}\\ \text{returns}}} & \underbrace{\hphantom{-(-0.039)}}_{\substack{\text{ABC ETF}\\ \text{returns}}} \end{array}\right] = \left[\begin{array}{r} \vdots \\ -0.002 \\ 0.003 \\ -0.807 \\ 0.614 \\ 0.189 \\ \vdots \\ \underbrace{\hphantom{-0.002}}_{\substack{\text{Return}\\ \text{spreads}}} \end{array}\right]. \tag{9}$$

In the first two periods, the returns of the synthetic index portfolio are almost the same as the returns to the ETF, producing only small deviations on the right-hand side. Here, only microstructure noise separates the two prices. In the third period, the ETF price jumps by 0.807 percent, while the prices of the individual stocks do not move much, leading to a large negative spread. In the fourth and fifth periods, the prices of the portfolio of individual stocks catch up to the ETF jump, leading to large positive spreads.

This stylized example illustrates the central problem in the analysis of common jumps on a fine sampling grid. If news reached the entire market instantly, was interpreted homogeneously, and trading were continuous, jumps in individual stocks would presumably occur simultaneously with the ETF index jump and the spreads would be small and random. Heterogeneous reactions to news produce asynchronous jumps in observed prices, however, and the spread temporarily expands and contracts again.

2.2.2 Decomposition of the spread

To isolate the effect of asynchronous jumps on the spread (6), we separate the return of the synthetic stock index portfolio into discontinuous (i.e., jump) and continuous parts. The return of this portfolio, $\Delta^n_i S$, is the weighted average of the observed returns from each individual stock in the portfolio, $\sum_{k=1}^{p} w_{k,i\Delta_n}\Delta^n_i Y_k$, in which $\Delta^n_i Y_k$, for $k = 1, ..., p$, is the Δ^n_i-th observed return of the k-th stock.

Jump tests, such as the one developed by Lee and Mykland (2008), help us flag large observed stock returns as jumps. We use Boudt and Zhang (2015)'s modified Lee and Mykland (2008) jump test that accounts for periodicity and microstructure noise. The Online Supplement details this test.

Define the binary variable $\text{Jump}_{k,i\Delta_n}$ as taking the value 1 when a jump is detected in the Δ^n_i-th return of the k-th stock and 0 otherwise. Each stock return can be decomposed as

$$\Delta^n_i Y_k = \Delta^n_i J_k + \Delta^n_i C_k, \tag{10}$$

i.e., the sum of a jump return, defined as $\Delta_i^n J_k := \Delta_i^n Y_k \cdot \text{Jump}_{k,i\Delta_n}$ and a continuous return $\Delta_i^n C_k := \Delta_i^n Y_k \cdot (1 - \text{Jump}_{k,i\Delta_n})$.

Using this decomposition, the return spread (6) is a linear combination of the weighted stock jump returns, the weighted continuous stock returns, and the ETF returns:

$$\delta^r_{i\Delta_n} \overset{(6)}{:=} \Delta_i^n S - \Delta_i^n Z \overset{(10)}{=} \sum_{k=1}^{p} \underbrace{w_{k,i\Delta_n} \Delta_i^n J_k}_{\substack{\text{Weighted} \\ \text{jump} \\ \text{stock returns}}} + \underbrace{\sum_{k=1}^{p} \underbrace{w_{k,i\Delta_n} \Delta_i^n C_k}_{\substack{\text{Weighted} \\ \text{continuous} \\ \text{stock returns}}} - \underbrace{\Delta_i^n Z.}_{\text{ETF returns}}}_{\text{Target}} \tag{11}$$

We seek to optimally change the time labels of the jump returns of individual stocks (the first term) to offset the ETF jump (included in the second term). The target value of the offset is the weighted sum of the continuous stock returns less the ETF return. Recall that if the stocks and the ETF impound news simultaneously, the spread in returns should be small, containing only microstructure noise.

Our method does not differentiate between ETF jumps originating from market-wide news or from a jump in a large-weight constituent stock. The method only requires that the ETF price path serves as a reliable target for the efficient impoundment of news. This point is particularly relevant for the price-weighted DJIA ETF in our empirical application.

2.2.3 Construction of the jump-event matrix

To synchronize jumps within an event window, we create a jump-event matrix Γ, which is an easier-to-handle representation of the decomposition in (11) with the following structure:

$$\Gamma = (\gamma_{il}) := \Big[\underbrace{w_{i\Delta_n} \Delta_i^n J}_{\substack{\text{Weighted} \\ \text{jump} \\ \text{stock returns}}}, \ \underbrace{T_{i\Delta_n}}_{\text{Target}} \Big]_{i \in \mathcal{W}_n}. \tag{12}$$

The matrix Γ consists of elements γ_{il}, with $i = 1, ..., h$ and $l = 1, ..., q$, where h is the number of time periods in the event window around the ETF jump, and $q-1$ is the number of jumps in individual stocks within the time interval, not the number of stocks. Some stocks may jump more than once or not at all. The event window, $\mathcal{W}_n := [I_1\Delta_n, I_2\Delta_n]$, covers a range of $h := I_2\Delta_n - I_1\Delta_n + 1$ time periods.

The matrix's first $q - 1$ columns consist of vectors of weighted stock jump returns sampled within this event window. If a stock jumps multiple times, such as stock A in our stylized example, separate jumps appear in different columns. Each jump vector contains one and only one non-zero element. Stocks that do not jump, such as stock C in our stylized example, do not contribute to the weighted stock jump returns. The matrix's qth column is the target vector, $T_{i\Delta_n} := (\sum_{k=1}^{p} w_{k,i\Delta_n} \Delta_i^n C_k) - \Delta_i^n Z$, which is the difference between the continuous returns of the synthetic stock index portfolio, $\sum_{k=1}^{p} w_{k,i\Delta_n} \Delta_i^n C_k$, and the ETF returns $\Delta_i^n Z$.

Within Γ, we rearrange the stock jumps, but never the elements of the target column. Each row-sum is a linear combination of stock jumps, the stock's continuous returns and the ETF returns, that equals the return spreads (11):

$$\Gamma^{+} := \sum_{m=1}^{q} (\gamma_{im}) = [\delta^{r}_{i\Delta_n}]_{i \in \mathcal{W}_n}. \tag{13}$$

In our ABC example, the jump-event matrix (12) is:

$$\Gamma = \left[\begin{array}{cccc} 0.000 & 0.000 & 0.000 & -0.002 \\ 0.000 & 0.000 & 0.000 & 0.003 \\ 0.000 & 0.000 & 0.000 & -0.807 \\ \underline{0.210} & 0.000 & \underline{0.400} & 0.004 \\ 0.000 & \underline{0.217} & 0.000 & -0.028 \end{array}\right]_{\underbrace{\hphantom{0.000\quad 0.000\quad 0.000}}_{\substack{\text{Weighted}\\ \text{jump}\\ \text{stock returns}}}\ \underbrace{\hphantom{-0.028}}_{\text{Target}}}, \text{ with row-sums } \Gamma^{+} = \begin{bmatrix} -0.002 \\ 0.003 \\ -0.807 \\ 0.614 \\ 0.189 \end{bmatrix}.$$

The first three columns of the jump-event matrix correspond to the three individual stock jumps across an event window from two minutes before to two minutes after the ETF jump. In this example, the first two columns contain the gradual jumps of stock A, and the third column contains the delayed jump of stock B. Generally, the stock jump sizes are weighted by their shares of the index, but this example uses an equally weighted index for simplicity. The fourth column in the jump-event matrix is a target vector that contains the difference between the continuous returns of the stocks and the ETF returns. The return spread is the sum of the row-sums of the weighted jump stock returns and the target column, *i.e.* the row-sums of the jump-event matrix (13). As before (in 2.2.1), the return spread becomes negative in period 3 and then positive in periods 4 and 5 because the ETF jumps in period 3, but stocks A and B do not jump until periods 4 and 5.

2.3 Rearranging the elements within the jump-event matrix

After decomposing the stock returns into jump and non-jump returns, we can rearrange the stock jumps in the jump-event matrix to offset the target column and minimize the variability of the return spreads, i.e., the row-sums.

A rearrangement of a specific column in the $h \times q$ jump-event matrix is determined by a permutation π_l of its h elements, with $l = 1, ..., q$. This permutation π_l is represented compactly as a vector mapping the original order into a new order:

$$\pi_l := \begin{pmatrix} 1 & 2 & ... & h \\ \pi_l(1) & \pi_l(2) & ... & \pi_l(h) \end{pmatrix}. \tag{14}$$

Each column in the jump-event matrix has a permutation. The collection of these permutations for all q columns is denoted by $\pi := (\pi_1, ..., \pi_q)$.

A rearrangement of the jump-event matrix, as defined in (12), is a new matrix, denoted with a π in its superscript:

$$\Gamma := \Big[\underbrace{w_{i\Delta_n} \Delta_i^n J}_{\substack{\text{Weighted}\\ \text{jump}\\ \text{stock returns}}},\ \underbrace{T_{i\Delta_n}}_{\text{Target}}\Big]_{i \in \mathcal{W}_n} \xrightarrow{\text{Rearrangement}} \Gamma^{\pi} := \Big[\underbrace{w_{i\Delta_n} \Delta_i^n J^{\pi}}_{\substack{\text{Rearranged}\\ \text{weighted}\\ \text{jump}\\ \text{stock returns}}},\ \underbrace{T_{i\Delta_n}}_{\text{Target}}\Big]_{i \in \mathcal{W}_n}. \tag{15}$$

In this new matrix, the elements γ^{π}_{il} correspond to the new configuration of the matrix. The vector $w_{i\Delta_n} \Delta_i^n J^{\pi} := (w_{1,i\Delta_n} \Delta_i^n J_1^{\pi}, ..., w_{q-1,i\Delta_n} \Delta_i^n J_{q-1}^{\pi})$ consists of the *rearranged* weighted stock jump returns. By changing the order of elements within the jump return vectors, this new version of the matrix reflects a different arrangement of stock jumps. The target column remains unchanged: we do not swap any elements in the qth column in the jump-event matrix, and it does not carry a π in its superscript.

The row-sums of the rearranged jump-event matrix Γ^{π} are the new return spreads:

$$\Gamma^{\pi,+} := \sum_{m=1}^{q} (\gamma_{im}^{\pi}). \tag{16}$$

The arrangement (and thus timing) of the stock jumps influence these row-sums. This makes them valuable for examining scenarios and understanding the impact of different timings.

Returning to the ABC example: let's consider a specific permutation π_1 (see (14)), which swaps the 3rd and the 4th observations in the first column in the jump-event matrix:

$$\pi_1 = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 2 & \underline{4} & \underline{3} & 5 \end{pmatrix}.$$

This swap rearranges the jump-event matrix, switching the 3rd and 4th rows of the first column, shifting a stock A's first jump back one period and a zero forward, thereby reducing the variability of the row-sums:

$$\Gamma^{\pi} = \left[\underbrace{\begin{matrix} 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 \\ \underline{0.210} & 0.000 & 0.000 \\ 0.000 & 0.000 & \underline{0.400} \\ 0.000 & \underline{0.217} & 0.000 \end{matrix}}_{\substack{\text{Rearranged} \\ \text{weighted} \\ \text{jump} \\ \text{stock returns}}} \; \underbrace{\begin{matrix} -0.002 \\ 0.003 \\ -0.807 \\ 0.004 \\ -0.028 \end{matrix}}_{\text{Target}} \right], \text{with row-sums } \Gamma^{\pi,+} = \begin{bmatrix} -0.002 \\ 0.003 \\ -0.597 \\ 0.404 \\ 0.189 \end{bmatrix}.$$

2.4 The best rearrangement of the jump-event matrix

Under the assumption that the latent prices of the stock index and the ETF move in lockstep, the best jump rearrangement minimizes the variability of the return spreads, i.e., misalignment. This reduction is called “flattening”. Formally, we minimize the return-spread range:

$$\min_{\pi} R\left(\Gamma^{\pi,+}\right). \tag{17}$$

The range $R(\cdot)$ is the difference between the highest and lowest row-sums:

$$R\left(\Gamma^{\pi,+}\right) = \Gamma_{(h)}^{\pi,+} - \Gamma_{(1)}^{\pi,+}, \tag{18}$$

in which the subscripts (i) in parentheses indicate the i-th order statistic of the sample of row-sums. By minimizing R, the optimization problem finds the best alignment of individual stock jumps with the ETF jump and minimizes the return spreads across all rows, reflecting a coordinated market reaction.

The combinatorial optimization problem (17) is studied by the pioneering work of Puccetti and Rüschendorf (2012) and Embrechts et al. (2013). Their Rearrangement

Algorithm loops over each column in a matrix to order it oppositely to the sum of the other columns (see Appendix B in the Online Supplement for details). This algorithm is best known as an actuarial tool to bound portfolio risk, but also has applications in other disciplines, such as option pricing (*e.g.*, Bondarenko and Bernard, 2024) and operations research (*e.g.*, Boudt et al., 2018).

The Rearrangement Algorithm does not constrain the rearrangements, however. For example, it can move jumps either forward or backward in time. We introduce the "Rearrangement Mixed-Integer Linear Program" (RMILP) to constrain the procedure from economically implausible rearrangements.

2.4.1 Understanding the RMILP

In operations research, one often seeks to minimize a function subject to a set of constraints. For example, one may wish to find a vector $\boldsymbol{x}$ that minimizes $\boldsymbol{c}^\top \boldsymbol{x}$, with $\boldsymbol{c}$ a given vector, subject to some constraints on $\boldsymbol{x}$. The RMILP introduced here follows the same principle, adding restrictions to rule out economically implausible rearrangements. To avoid profligacy – that is, excessive or unrealistic changes to the original data – we solve the optimization problem under progressively relaxed constraint, allowing just one, two, or more shifts in time.

Permutations as permutation matrices

To rearrange (i.e., permute) the h observations in a single column in the jump-event matrix, we use the column representation of a permutation matrix. Given that the jump-event matrix has 4 columns, our solution will involve 4 permutation matrices, one for each column.

The $h \times h$ permutation matrix, denoted as P_{π_l}, reorders the columns of the identity matrix I_h to represent a permutation π_l, as we defined earlier in (14):

$$P_{\pi_l} = (p_{ii'}) = \begin{bmatrix} \boldsymbol{e}_{\pi_l(1)} \\ \boldsymbol{e}_{\pi_l(2)} \\ \vdots \\ \boldsymbol{e}_{\pi_l(h)} \end{bmatrix}. \tag{19}$$

For each i, the element $p_{ii'}$ is 1 if $i' = \pi_l(i)$, and 0 otherwise. This means that the entries of the ith row are all zero except for a 1 that appears in column $\pi_l(i)$. We can think of these rows as basis vectors, where $\boldsymbol{e}_{i'}$ is a row-vector of length h with a 1 on position i' and a 0 on every other position. Note that the arguments in the subscripts of the basis vectors match the permutation order in (14).

This representation is crucial for the RMILP because a permutation matrix (19) can track how far each element shifted by measuring deviations from the diagonal. Such tracking is essential for imposing economically meaningful restrictions on movement: some jumps may be constrained to move only within a limited range, while others (such as those already synchronized with the ETF) are not allowed to move at all. For example, we can specify a maximum allowable distance from the diagonal to not let the jumps stray too far within the event window (see Section 2.4.3 for further elaboration).

The permutation discussed in the example of Section 2.3, π_1 switches the 3rd and 4th rows of the 1st column in a jump-event matrix, and corresponds to the following 5×5

(there are 5 rows in the jump-event matrix) permutation matrix:

$$P_{\pi_1} = \begin{bmatrix} \boldsymbol{e}_{\pi_1(1)} \\ \boldsymbol{e}_{\pi_1(2)} \\ \boldsymbol{e}_{\pi_1(3)} \\ \boldsymbol{e}_{\pi_1(4)} \\ \boldsymbol{e}_{\pi_1(5)} \end{bmatrix} = \begin{bmatrix} \boldsymbol{e}_1 \\ \boldsymbol{e}_2 \\ \boldsymbol{e}_4 \\ \boldsymbol{e}_3 \\ \boldsymbol{e}_5 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \underline{1} & 0 \\ 0 & 0 & \underline{1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

where column i' of the I_5 identity matrix now appears as the column $\pi(i')$ of P_{π_1}. The changes occur in the vertical, i, dimension: upward moves in the permutation matrix are backward moves in time, while downward moves in the permutation matrix are forward moves in time. The fourth element, which is on position $(4, 4)$ in I_5, shifts one spot backward in time by shifting one step upward in the permutation matrix. The third element, which is on position $(3, 3)$ in I_5, shifts one step forward in time by shifting one step downward in the permutation matrix. In other words, P_{π_1} switches the period 3 return with the period 4 return.

There is a distinct permutation matrix for each of the q columns in the jump-event matrix, i.e., for each stock jump and the target. These matrices are then concatenated to form the co-permutation matrix Π of dimension $h \times (hq)$:

$$\Pi = (p_{lii'}) = \left[P_{\pi_1}, P_{\pi_2}, ..., P_{\pi_q}\right], \tag{20}$$

with $l = 1, ..., q$, $i, i' = 1, ..., h$ and P_{π_1} is short notation for the $h \times h$ permutation matrix for the first column in the jump-event matrix. Similarly, $P_{\pi_2}, P_{\pi_3}, ..., P_{\pi_q}$ are the permutation matrices for the 2nd, 3rd, ..., and qth columns, respectively.

Extracting the row-sums with permutation matrices

To produce the return spreads, we transform the jump-event matrix into a column vector and then premultipy the (vectorized) jump-event matrix by the co-permutation matrix:

$$\Gamma^{\pi,+} = \Pi \times \text{vec}(\Gamma), \tag{21}$$

or, equivalently:

$$\begin{bmatrix} J^{\pi,+}_{n,1} \\ J^{\pi,+}_{n,2} \\ J^{\pi,+}_{n,3} \\ \vdots \\ J^{\pi,+}_{n,h} \end{bmatrix} = \left[\begin{array}{cccc|cccc|c|cccc} p_{111} & p_{112} & \cdots & p_{11h} & p_{211} & p_{212} & \cdots & p_{21h} & \cdots & p_{q11} & p_{q12} & \cdots & p_{q1h} \\ p_{121} & p_{122} & \cdots & p_{12h} & p_{221} & p_{222} & \cdots & p_{22h} & \cdots & p_{q21} & p_{q22} & \cdots & p_{q2h} \\ \vdots & & \ddots & \vdots & \vdots & & \ddots & \vdots & & \vdots & & \ddots & \vdots \\ p_{1h1} & p_{1h2} & \cdots & p_{1hh} & p_{2h1} & p_{2h2} & \cdots & p_{2hh} & \cdots & p_{qh1} & p_{qh2} & \cdots & p_{qhh} \end{array}\right] \times \begin{bmatrix} \gamma_{11} \\ \gamma_{21} \\ \vdots \\ \underline{\gamma_{h1}} \\ \vdots \\ \gamma_{hq} \end{bmatrix},$$

in which vec(Γ), the vectorized version of the observed jump-event matrix $\Gamma = (\gamma_{il})$ with $i = 1, ..., h$ and $l = 1, ..., q$, is a stacked column vector of dimension $hq \times 1$. The co-permutation matrix Π rearranges this vector. The matrix product in (21) is an $h \times 1$ column vector (the left side of the equal sign). Each entry is a row-sum from the rearranged jump-event matrix.

Flattening the row-sums by choosing the best permutation matrices

Recall that linear programs are problems that can be expressed in canonical form as: "Find a vector $\boldsymbol{x}$ that minimizes $\boldsymbol{c}^\top \boldsymbol{x}$, with $\boldsymbol{c}$ a given vector, subject to some constraints on $\boldsymbol{x}$". Because we cannot directly express the objective function, R, as a function of the permutation matrices in Π, the RMILP steps slightly outside the bounds of conventional linear programming to get the appropriate form indirectly.[7] The RMILP aims to find the co-permutation matrix Π, along with two auxiliary boundary values, L (lower) and U (upper), to minimize the interval $U - L$:

$$\begin{aligned} &\text{Find } \Pi, L, U \text{ that minimizes } U - L, \\ &\qquad\qquad \text{subject to } L \leq J_{n,i}^{\pi,+} \leq U, \text{ for } i = 1, ..., h, \end{aligned} \tag{22}$$

in which Π is the co-permutation matrix, which defines the rearranged jump-event matrix, and L and U define lower and upper bounds of an unknown interval.

This objective function alone does not directly minimize the range of the row-sums. We link L and U to Π by constraining the decision variables with inequality constraints: L and U mark the bounds within which all row-sums $J_{n,i}^{\pi,+}$ fall. Seemingly trivial, this constraint indirectly sets the smallest possible (the minimum) and largest possible (maximum) row-sum.[8] By minimizing $U - L$, subject to the inequality constraint, the RMILP squeezes these boundaries together, as closely as possible, and thereby ensures that the optimal solutions for L and U (denoted L^* and U^*) equal the smallest and largest row-sums:

$$L^* = J_{n,(1)}^{\pi,+} \text{ and } U^* = J_{n,(h)}^{\pi,+}. \tag{23}$$

A simple proof by contradiction suffices to confirm this statement: if L^* and U^* did not correspond to the smallest and largest row-sums, one could still further reduce $U - L$.

The program in (22) chooses three decision variables – Π, L, and U– to minimize the range of the row-sums. The optimization problem thus becomes a "mixed-integer" linear program, with Π composed of binary elements $(0, 1)$ and continuous values for L and U.

2.4.2 Other necessary constraints for the RMILP

In the RMILP, we also restrict the co-permutation matrix (20) with a permutation constraint and a target constraint. The permutation constraint requires that each permutation matrix is composed entirely of zeros and ones, and that each row and column in the permutation matrix sums to one. An identity matrix satisfies this constraint, for example, as does an identity matrix with its rows rearranged in any order. The target constraint prohibits changes in the order of the elements in the target column. The combination of these minimal conditions produces rearrangements that minimize the range.

[7] While the matrix product in (21) can, with a fixed co-permutation matrix Π, easily extract any specific row-sum, i.e., $J_{n,1}^{\pi,+}, J_{n,2}^{\pi,+}, ..., J_{n,h}^{\pi,+}$ (indices without brackets), there is no possible choice of Π that produces the maximum or minimum of the row-sums specified in the objective function, i.e., the order statistics $J_{n,(1)}^{\pi,+}$ and $J_{n,(h)}^{\pi,+}$ (indices with brackets) through the matrix product alone.

[8] The row-sums $J_{n,i}^{\pi,+}$ in the middle of the constraint result from the matrix product in (21) for a given arrangement of jumps. The left inequality in the constraint of (22) defines a lower boundary in a set of row-sums (by definition, each individual row-sum should greater than or equal to the minimum) and the right inequality defines an upper boundary in a set of row-sums (by definition, each individual row-sum should be less than or equal to the maximum). The outer boundaries, the lower boundary should be less than or equal to the upper boundary, ensures that the minimum is always smaller than the maximum, as by definition.

The permutation constraints

To define a proper co-permutation matrix (20), we set up equality constraints in the linear program:

$$\text{subject to } \sum_{l=1}^{q}\sum_{i=1}^{h}\sum_{i'=1}^{h} p_{lii'} = hq, \tag{24}$$

$$\sum_{i'=1}^{h} p_{lii'} = 1, \text{ for } i = 1, ..., h \text{ and } l = 1, ..., q, \tag{25}$$

$$\sum_{i=1}^{h} p_{lii'} = 1, \text{ for } l = 1, ..., q \text{ and } i' = 1, ..., h. \tag{26}$$

Constraint (24) requires that each permutation matrix (19) contains h ones, corresponding to selecting all (exactly h) elements in each column in the jump-event matrix. Thus, the total sum across all q permutation matrices in the co-permutation matrix equals hq. Constraint (25) imposes that the rows of each permutation matrix to sum to 1. This prevents multiple or no elements in certain positions of the rearranged matrix. Finally, constraint (26) guarantees that no number is repeated within any column in the rearranged matrix.

The target

We also impose an equality constraint on the permutation matrix of the last column in the jump-event matrix P_{π_q} that ensures that the RMILP does not rearrange the target column:

$$\text{subject to } (\text{diag}(P_{\pi_q}))_i = 1, \text{ for } i = 1, \ldots, h. \tag{27}$$

In other words, the permutation matrix (19) associated with the qth column in the jump-event matrix P_{π_q} must be an identity matrix.

2.4.3 Adapting the RMILP with additional constraints

Linear programming allows us to rule out economically implausible rearrangements of jumps. For example, we can prohibit large moves or restrict moves in a particular direction with the aid of a distance matrix, denoted D_n. This matrix tracks how far each element in a permutation matrix (19) is displaced from its original position. Because the baseline order of a permutation matrix is an identity matrix of dimension $h \times h$, D_n is a matrix of the same dimension, with zeros on the diagonal and increasing values away from the diagonal:

$$D_n = (d_{ii'}) \begin{bmatrix} 0 & 1 & \cdots & h-2 & h-1 \\ 1 & 0 & \cdots & h-3 & h-2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ h-2 & h-3 & \cdots & 0 & 1 \\ h-1 & h-2 & \cdots & 1 & 0 \end{bmatrix}. \tag{28}$$

In practice, we do not need the full distance matrix. We simplify it by restricting attention to economically meaningful movements. In this paper, we only allow stock jumps to be moved backward in time because we assume that stock prices cannot plausibly lead the highly liquid and carefully watched ETF. To only monitor these backward moves, we use the upper triangular portion of D_n and set the remaining entries to zero. Moreover, because the permutation matrix (19) tracks the rearrangements of both jumps and non-jumps (the zero entries in the jump-event matrix), we disable irrelevant columns of D_n, so that only jump positions contribute to the computed distances. That is, if a price jump occurs in period $i = i^*$, we set all columns with $i' \neq i^*$ of D_n to zero.

The distance matrix provides a simple way to measure how far each jump is moved in time. For each column l in the jump-event matrix, we can calculate the total distance traveled as a matrix product of a vectorized permutation matrix (encoding the rearrangement) and a vectorized distance matrix (quantifying the displacement implied by that rearrangement):

$$d(P_{\pi_l}) = \text{vecr}\left(P_{\pi_l}\right) \times \text{vecr}\left(D_n\right)^\top, \text{ for } l = 1, ..., q. \tag{29}$$

The vecr$(\cdot)$ function concatenates the rows of a matrix (unlike standard vectorization, which stacks columns), producing a $1 \times hq$ row-vector. Having quantified the distance each jump can move, we can now impose a constraint that limits it:

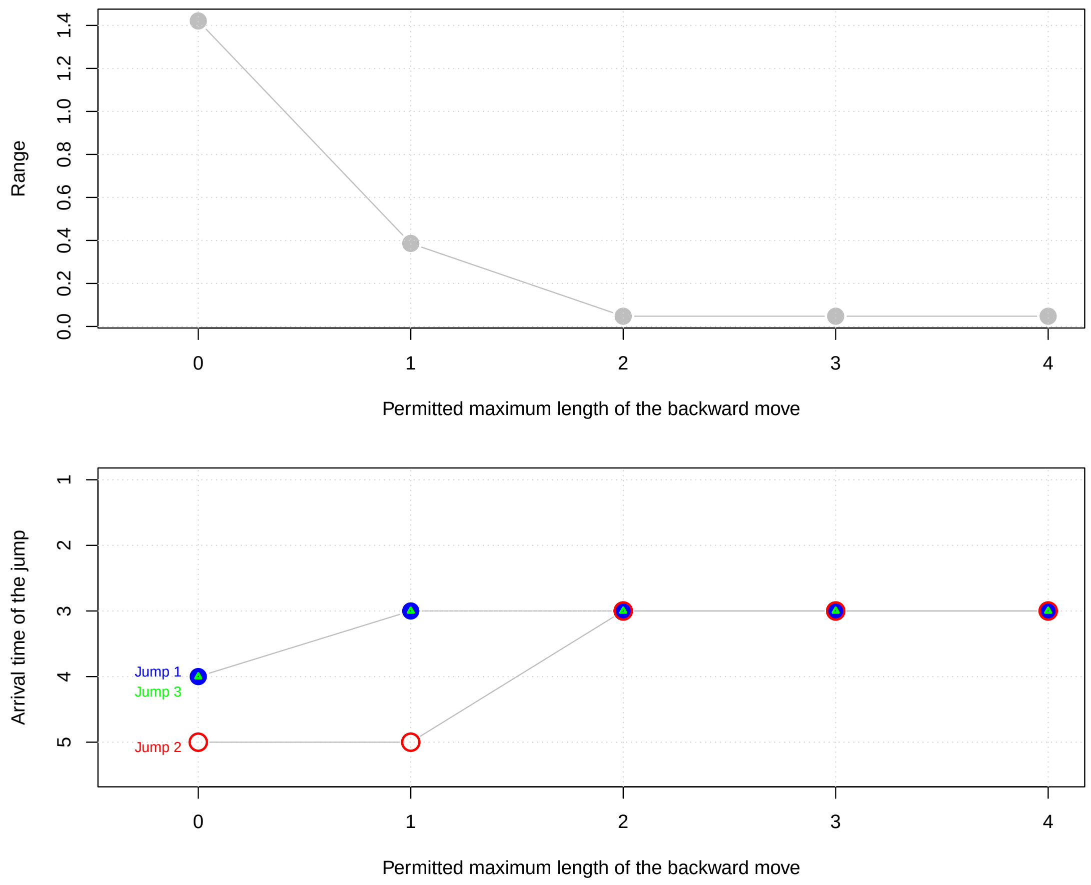
$$\text{subject to } d(P_{\pi_l}) \leq c, \text{ for } l = 1, ..., q-1, \tag{30}$$

where $c > 0$ denotes the maximum permitted backward shift. The qth column is exempt from this constraint, as the target column remains fixed. In applying our methods to the data, we re-optimize under several candidate values of c and select the solution that achieves the smallest criterion value with the fewest rearrangements. In practical applications, the constraint could also be tailored to each stock based on its liquidity.

The trace plot in Figure 3 illustrates how the RMILP behaves as we relax the constraint on the maximum allowable length of the move. The top panel shows the range of the spreads (i.e., the criterion value), as a function of the permitted maximum length of the move – set to a maximum of zero, one, two, three or four minutes – for the jump-event matrix from the previous stylized example. The negative slope of the line shows that permitting larger backward shifts progressively flattens the return spreads, indicating closer synchronization across assets. The bottom panel shows how corresponding jump arrival periods change with the constraint. With no backward moves allowed, jumps 1 and 3 arrive in period 4 and jump 3 arrives in period 5, as in the observed prices. If we permit one backward move for each jump, the RMILP moves jumps 1 and 3 to period 3 while if we permit 2 backward moves, the RMILP moves all jumps to period 3. The smallest range (0.048 percent) occurs with two backward shifts, aligning all the jumps in the third period in the bottom panel. For reference, this optimal rearrangement produces the following transformation of the jump-event matrix, with the jumps now aligned in period 3:

$$\Gamma = \left[\begin{matrix} \underbrace{\begin{matrix} 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 \\ \underline{0.210} & 0.000 & \underline{0.400} \\ 0.000 & \underline{0.217} & 0.000 \end{matrix}}_{\substack{\text{Weighted} \\ \text{jump} \\ \text{stock returns}}} & \underbrace{\begin{matrix} -0.002 \\ 0.003 \\ -0.807 \\ 0.004 \\ -0.028 \end{matrix}}_{\text{Target}} \end{matrix} \right] \xrightarrow[\text{rearrangement}]{\text{The best}} \Gamma^\pi := \left[\begin{matrix} \underbrace{\begin{matrix} 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 \\ \underline{0.210} & \underline{0.217} & \underline{0.400} \\ 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 \end{matrix}}_{\substack{\text{Rearranged} \\ \text{weighted} \\ \text{jump} \\ \text{stock returns}}} & \underbrace{\begin{matrix} -0.002 \\ 0.003 \\ -0.807 \\ 0.004 \\ -0.028 \end{matrix}}_{\text{Target}} \end{matrix} \right].$$

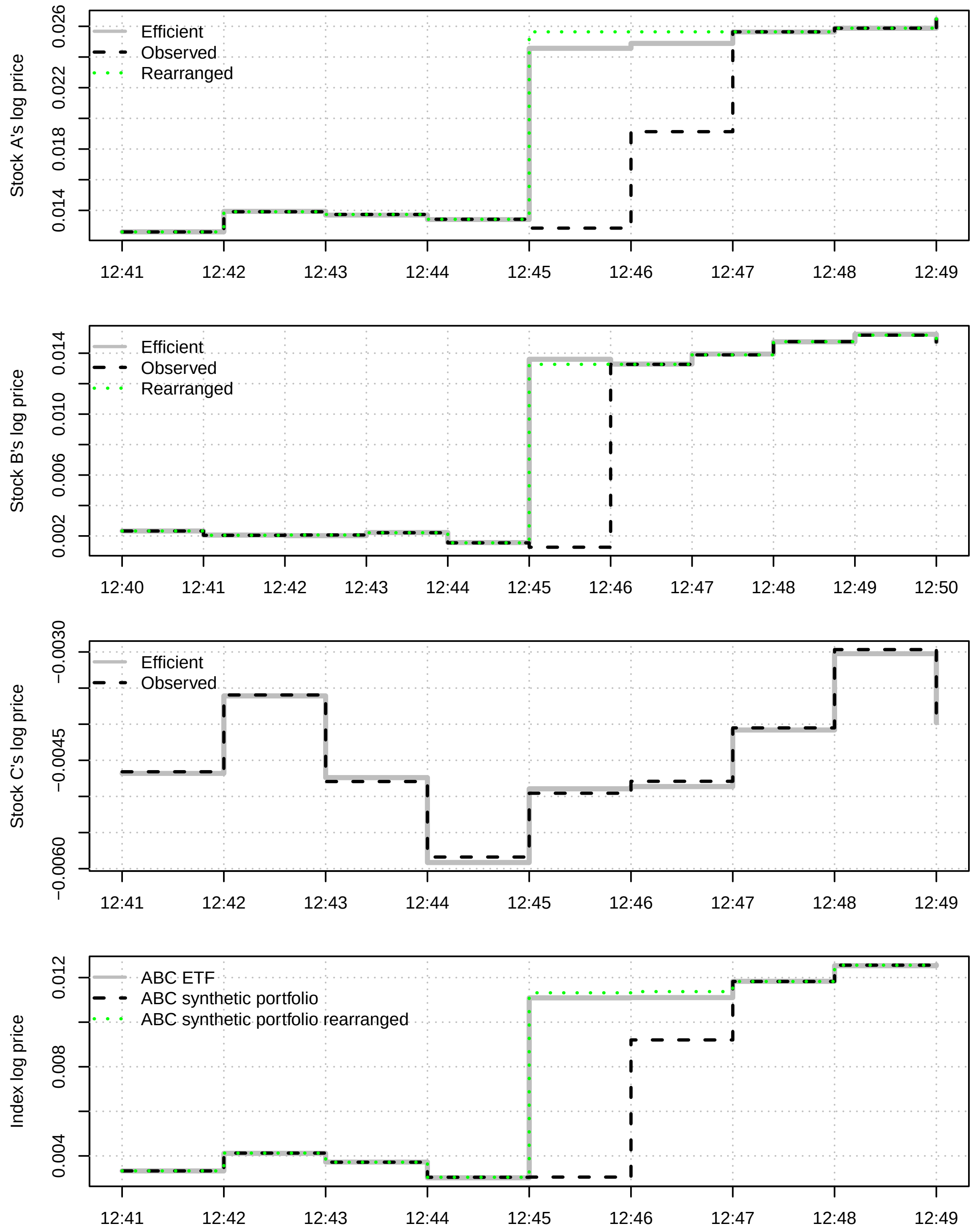
Figure 3: Trace plot – Range of the spreads and jump arrival times as a function of the permitted maximum length of the move



Note: This figure plots the range of the return spreads (the row-sums of the jump-event matrix) and the implied jump arrival times as a function of the length of permitted backward moves of the jumps in each of the columns. The RMILP rearranges the gradual jump of stock A (Jump 1 in the 4th period and Jump 2 in the 5th period) and the delayed jump of stock B (Jump 3 in the 4th period). We allow each jump to move backward in time a maximum of zero, one, two, three, four minutes and solve the RMILP for each of these five possible constraints.

Figure 4 shows prices before and after this optimal rearrangement of jumps. The first and second panels show that the observed (black) prices of stock A and stock B deviate from their efficient (gray) values after the ETF jump at 12:45. The third panel shows very small deviations in stock C's observed prices due to the contamination of the continuous component. The fourth panel shows that the delays cause the implied (inefficient) price of the basket of stocks (the ABC portfolio) to deviate noticeably from the efficient ABC ETF price from 12:45 to 12:47. The fourth panel also shows that the best rearrangement combines two small jumps of stock A to period 3 and shifts the jumps of stock B one period backward to period 3, aligning the jumps in time. The rearranged price paths (green) are now much closer to the efficient price paths (black) in the 1st, 2nd and 4th panels.

Figure 4: The best rearrangement approximately recovers the efficient stock jumps



Note: This figure shows the efficient (gray), observed (black) and rearranged (green) log prices of the assets. The top panel shows that stock A jumps gradually and completes the jump 2 minutes after the ETF in bottom panel, while the second panel shows that stock B's jump is 1 minute late and stock C does not jump at all (third panel). The fourth panel shows that the delays cause the implied (inefficient) price of the basket of stocks (the ABC portfolio) to deviate from the efficient ABC ETF price. The best rearrangement approximately recovers the latent, efficient stock price. That is, by shifting the jumps of stock A and stock B to earlier periods (top 2 panels), the RMILP minimizes the distance between the price of the ETF and the synthetic portfolio (lowest panel).

3 Empirical application

We apply our proposed method to study how individual stock prices react in event windows around ETF jumps. This exercise shows that timing frictions are common and that rearrangement recovers cross-sectional dependence that would otherwise require temporal aggregation. Jumps are associated with news and jump rearrangements are even more strongly associated with monetary policy announcements. We show that rearrangement improves the estimation of realized covariances and jump betas. Compared to raw returns, rearrangement reduces bias in estimated realized covariances, improves the predictive power of realized jump betas and provides superior inputs to trading rules.

3.1 Data: The Dow and the DIA ETF

The NYSE Trade and Quote (TAQ) database provides equity trade data with millisecond precision timestamps. The basket instrument is the SPDR Dow Jones Industrial Average ETF (DIA). We compare this ETF to the price of a synthetic index of Dow 30 stock prices, as in Bollerslev et al. (2008).[9]

The data span January 3, 2007 through April 2, 2020, which includes several exceptionally turbulent episodes, such as the global housing and credit crisis, the European sovereign debt crisis and the bail-out of Greece, the initial Russian invasion of Ukraine in 2014, the Turkish crisis of 2018, and the 2020 stock market crash.

We pre-filter the prices as in Barndorff-Nielsen et al. (2009). We also remove banking holidays, half-trading days, any day where there is more than a two-hour gap between consecutive trades and periods of malfunctioning markets, such as the 2010 flash crash, leaving a total of $N = 3,264$ trading days.

There are many jumps in the data. Following Lee and Mykland (2008) and Boudt et al. (2011b), a return is said to be contaminated by a jump when that return is abnormally high compared to a local and robust (jump-free) measure of volatility. We follow the methods of Boudt and Zhang (2015) that account for intraday periodicity and microstructure noise in computing the local measure of the volatility. Section C of the Online Supplement describes the test. We test for jumps using one-minute returns. (i.e., $1/\Delta_n = 390$). To minimize the problem of finding spurious jumps with many jump tests over a sample, we follow Lee and Mykland (2008) and set a critical value based on extreme value theory. That is, the critical values for our jump tests come from the distribution of maxima for a set of test statistics, not the distribution of a single test statistic. More specifically, the critical value is the 0.1% of the Gumbel distribution, which characterizes the maximum of a set of M i.i.d. realizations of the absolute value of the standard normal random distribution. To control the family-wise error rate across the entire sample, we set $M = 1/\Delta_n \times N$. For $1/\Delta_n = 390$ one-minute returns and $N = 3,264$ days, this gives a critical value of 6.25.

Table 1 reports the number of jumps for the ETF and each stock in the DIA over 2007–2020. The jump test identifies 638 ETF jumps across 419 (jump) days (i.e., 12.8% of the sample are jump days). Some days include multiple, even consecutive, 1-minute ETF

[9] The constituency of the Dow 30 index is dynamic. The 43 unique tickers within our sample period are: AA, AAPL, AIG, AXP, BA, BAC, C, CAT, CSCO, CVX, DD, DIS, DOW, DWDP, GE, GM, GS, HD, HON, HPQ, IBM, INTC, JNJ, JPM, KFT, KO, MCD, MMM, MO, MRK, MSFT, NKE, PFE, PG, T, TRV, UNH, UTX, V, VZ, WBA, WMT and XOM.

jumps. Jump behavior varies substantially across firms: while most DIA stocks jump on 7–13% of days, some stocks (e.g., DuPont and General Motors) jump more often.

We construct 525 jump-event matrices (as in Equation (12)) around 638 ETF jumps. Each event window spans from 5 minutes before the ETF jump to 5 minutes after the ETF jump. When two or more jumps occur within 5 minutes of each other, we combine them into a single window that starts 5 minutes before the first jump and ends 5 minutes after the last jump. For example, if jumps occur at 2:16 and 2:20, the window starts at 2:11 and ends at 2:25. For this reason, there are more ETF jumps (638) than jump-event matrices (525). When two jumps occur more than 5 but less than 10 minutes apart, we create two separate, non-overlapping event windows with the end of the first window and start of the second window halfway between the two jumps. For example, if jumps occur at 2:16 and 2:25, the first window runs from 2:11 to 2:20 and the second runs from 2:21 to 2:30.

We constrain the RMILP in three ways. In every jump-event matrix:

1. We do not rearrange the price jumps of stocks that jump with the index, because we assume they are already efficient.
2. No stock price jumps may be moved earlier than the ETF jump.
3. No stock price jumps may be moved if the ETF jumps within the first 10 minutes or the last 10 minutes of the trading day.

After we impose these filters, 159 matrices remain as candidates for rearrangement. The RMILP rearranges stock jumps in 89 cases (or 17.0% of all jump-event matrices). That is, there are many asynchronously observed jumps. The last column in each panel of Table 1 shows that all stocks are rearranged at least once and some stocks exhibit many rearrangements (e.g., 24 for MMM).

In Figure 5, we show that rearrangement flattens the empirical return spreads for the DIA–Dow 30 stocks around ETF jumps, thus reducing misalignment. The left panel plots the highly skewed distribution of empirical ranges in 1-minute raw return spreads across the 159 event windows. For each event, the range is the difference between the highest and lowest return spreads, as defined in Equation (18), where the spreads themselves (defined in Equation (6)) are given by the row-sums of the jump-event matrix, as discussed in Section 2.2.3. The right panel shows how applying the RMILP reduces these ranges in 89 of 159 events, reducing the ranges by about 21% on average (marked by the red dotted line).

3.2 Jump rearrangements and macroeconomic news

We surmise that sluggish reactions to macroeconomic news are key drivers of mistimed jumps. To investigate this hypothesis, we match ETF jumps with a comprehensive database that covers over one hundred distinct types of U.S. macroeconomic and policy announcements. These events include monetary policy decisions (e.g., FOMC rate decisions, Fed bond purchase news, FOMC meeting minutes), the publication of business sentiment and manufacturing indicators (e.g., ISM Manufacturing, ISM Prices Paid, Philadelphia Fed Business Outlook), labor market statistics (e.g., nonfarm payrolls, unemployment rate, jobless claims), housing market indicators (e.g., building permits, housing starts, existing and new home sales), and price and inflation measures (e.g., CPI, PPI, PCE, Core PCE).

Table 1: Jumps in the DIA ETF and jumps and rearrangements in its constituents (2007–2020)

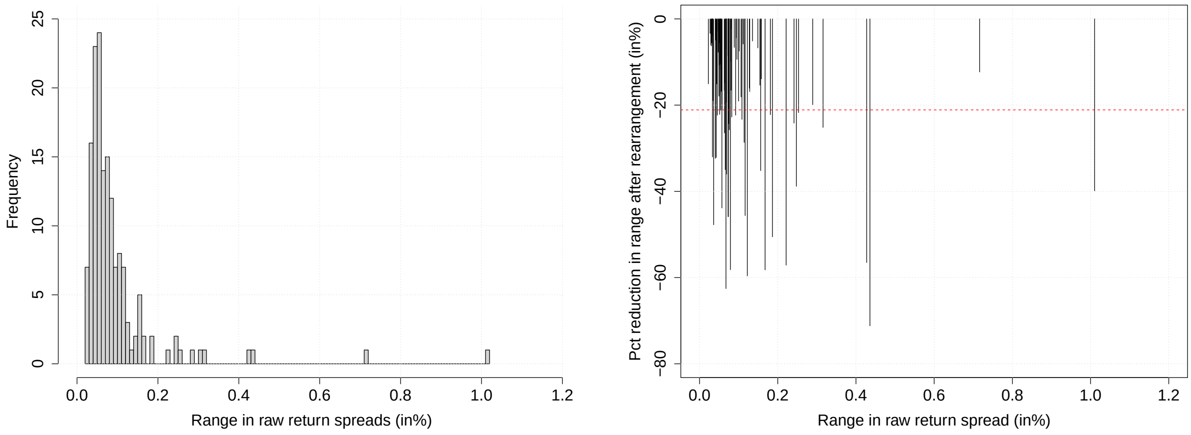
	Period	#Days	#J	#J-D	%-J-D	#RA
DIA	2007–2020	3,264	638	419	12.8	
Stocks:						
AA	2007–2013	1720	174	127	7.38	1
AAPL	2015–2020	1297	268	154	11.87	8
AIG	2007–2008	493	102	59	11.97	4
AXP	2007–2020	3264	471	288	8.82	12
BA	2007–2020	3264	631	376	11.52	5
BAC	2008–2014	1721	160	115	6.68	1
C	2007–2009	733	105	70	9.55	2
CAT	2007–2020	3264	386	275	8.43	12
CSCO	2009–2020	2771	288	211	7.61	3
CVX	2008–2020	3018	258	214	7.09	4
DD	2007–2020	2835	376	274	9.66	7
DIS	2007–2020	3264	348	247	7.57	5
DOW	2019–2020	247	82	52	21.05	3
DWDP	2017–2019	429	64	51	11.89	1
GE	2007–2018	2955	308	237	8.02	8
GM	2007–2009	595	271	144	24.2	2
GS	2013–2020	1789	302	194	10.84	8
HD	2007–2020	3264	355	236	7.23	3
HON	2007–2008	493	74	61	12.37	2
HPQ	2007–2013	1720	211	125	7.27	5
IBM	2007–2020	3264	456	325	9.96	13
INTC	2007–2020	3264	424	265	8.12	8
JNJ	2007–2020	3264	522	366	11.21	6
JPM	2007–2020	3264	367	243	7.44	14
KFT	2008–2012	1172	151	129	11.01	1
KO	2007–2020	3264	343	261	8.00	5
MCD	2007–2020	3264	449	316	9.68	4
MMM	2007–2020	3264	549	379	11.61	24
MO	2007–2008	493	149	85	17.24	1
MRK	2007–2020	3264	423	286	8.76	5
MSFT	2007–2020	3264	395	293	8.98	7
NKE	2013–2020	1789	266	189	10.56	3
PFE	2007–2020	3264	263	203	6.22	6
PG	2007–2020	3264	387	296	9.07	9
T	2007–2015	2214	297	211	9.53	5
TRV	2009–2020	2771	405	308	11.12	11
UNH	2012–2020	2034	335	228	11.21	2
UTX	2007–2020	3264	442	319	9.77	9
V	2013–2020	1789	305	201	11.24	6
VZ	2007–2020	3264	444	291	8.92	5
WBA	2018–2020	554	154	80	14.44	1
WMT	2007–2020	3264	428	283	8.67	4
XOM	2007–2020	3264	315	248	7.60	6

Note: The table reports descriptive statistics on ETF and constituent jumps for the Dow Jones Industrial Average (DIA) and its member stocks over the sample period 2007–2020. For each stock, we indicate the trading period, the number of trading days (#Days), the total number of detected jumps (#J), the number of jump days (#J-D), the jump day proportion (%-J-D), and the number of rearrangements (#RA).

Figure 5: DIA-Dow 30 return spreads flatten after rearrangement

(a) Histogram of empirical ranges

(b) Percentage reduction in empircal ranges



Note: Panel (a) plots the histogram of empirical ranges in raw return spreads across the 159 candidate events. Panel (b) shows the percentage reduction in these ranges after applying the rearrangement procedure. Of the 159 events, 89 involve a rearrangement, with an average reduction of approximately 21% (indicated by the red dotted line).

Of all 638 ETF jumps, 159 occur within $[-5, +10]$-minutes of a macroeconomic announcement (about 25% of all jumps).[10] But many more jumps are not obviously associated with any scheduled announcement or known cause, a puzzle that Lahaye et al. (2011) and Neely et al. (2025) also documented. The share of rearranged jump events associated with a news announcement is much higher than the analogous share of unrearranged jump events: 46% of rearranged jumps occur near announcements, compared with only 22% of unrearranged jump events.

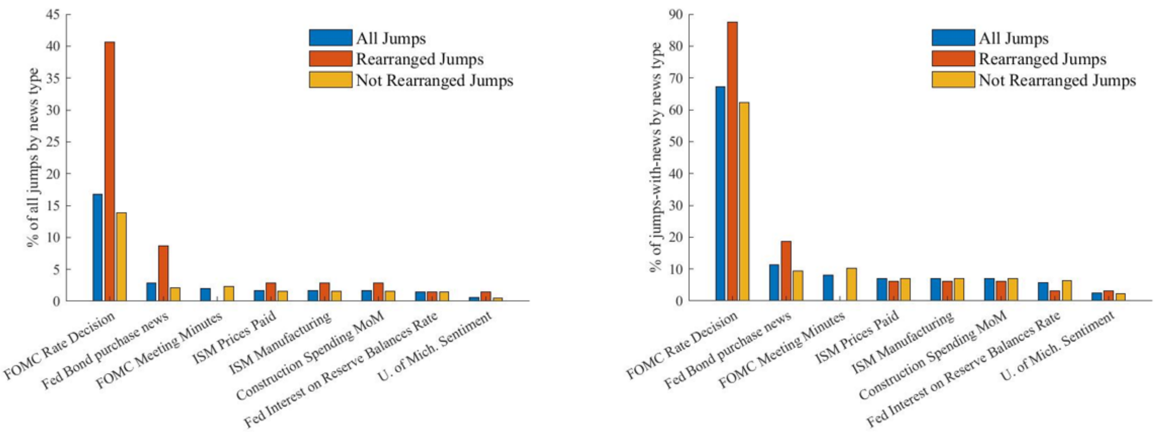
Rearranged jumps also involve more individual stock (co)jumps on average than unrearranged jumps. The difference is larger for jumps associated with news. Conditional on an announcement, there are 1,474 cojumps for rearranged jumps vs. 1,407 cojumps for unrearranged jumps, a 4.8% increase. For jumps that are not associated with an announcement, rearranged jumps are still associated with more cojumps than unrearranged jumps, but the difference is smaller (2.2%). This indicates that rearranged jumps are associated with a slightly stronger cross-sectional co-movement following news, suggesting that macroeconomic announcements tend to trigger more widespread jump activity across stocks.

Figure 6 breaks down how frequently the most important types of news are associated with ETF jumps. The left panel normalizes types of news associated with jumps by all jumps, while the right panel normalizes by jumps that occur within a $[-5, +10]$-minute window of a macroeconomic announcement. Both panels show that rearranged jumps are disproportionally linked to Federal Reserve communications. Nearly half of all rearranged jumps matrices (left panel) coincide with FOMC rate decisions or bond purchase

[10] We examined a window from 5 minutes before a jump time to 10 minutes after. We permit 5 minutes before news in case there was some minor mistiming of jumps or news. Indeed, allowing for jumps a few minutes before news appears to have been a sound practice. There were a few jumps in the 2 minutes before news, but no jumps 4 or 5 minutes before the news, suggesting that mistiming or news leakage was possible.

announcements, compared with roughly 15% among unrearranged jump matrices. When only jumps occurring near announcements are taken into account (right-hand panel), the difference becomes even more pronounced: more than 80% of rearranged jumps with news are tied to monetary policy events.[11] These findings support our interpretation that sluggish market reactions to macroeconomic and policy announcements contributes substantially to the mistiming we observe.

Figure 6: Association between macroeconomic announcements and ETF jumps


(a) All jumps

(b) Jumps associated with announcements

Note: The figure shows the share of jumps associated with specific types of macroeconomic announcements. The left panel expresses each category as a percentage of all ETF jumps, whether or not there is an announcement. The right panel expresses it as a percentage of jumps that occur within a $[-5; +10]$-minute window of an announcement ("jumps-with-news").

3.3 An empirical illustration for Sep. 18, 2007

The previous subsection confirms the well-established finding that macroeconomic news is correlated with asset price jumps, including those of ETFs (Andersen et al., 2003; Lahaye et al., 2011). A notable example is September 18, 2007, when the Federal Reserve announced a 50-basis-point cut in the federal funds target rate. The financial press widely described the decision as a bold action.[12] In this subsection, we use data from this event to illustrate how rearrangement can correct for sluggish stock-price reactions to a major news release.

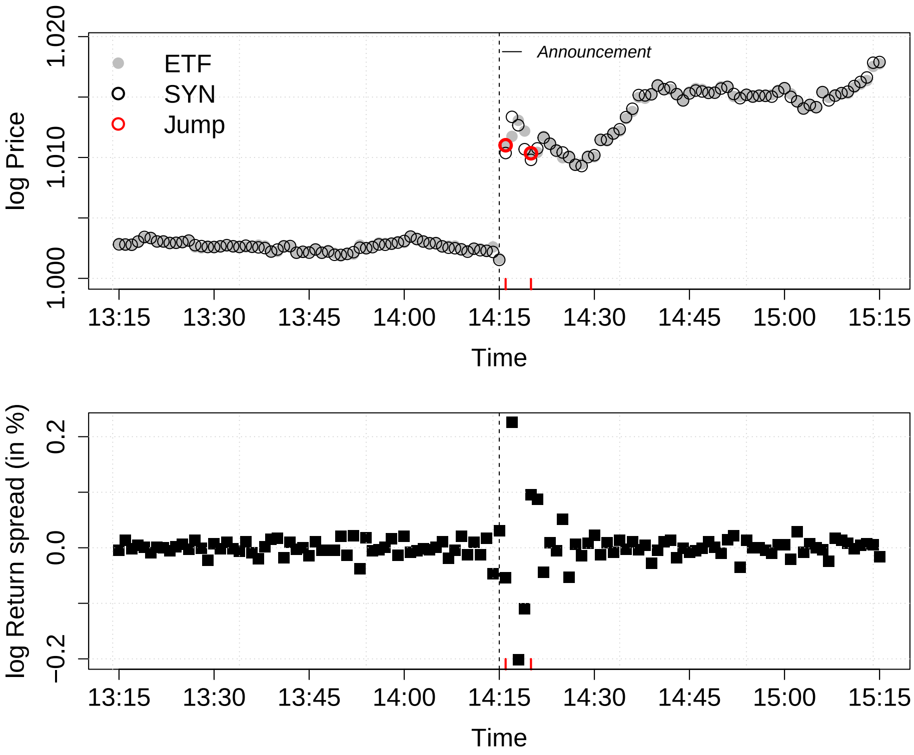
Figure 7 shows the price paths of the DIA ETF (in gray) and a synthetic Dow 30 index (in black), along with their return spreads (6) on the day of the announcement. The jump test flags a 0.938% ETF price jump at 2:16 pm, one minute after the release of the FOMC statement, followed by a smaller, –0.184% jump at 2:20 pm. For most of

[11] Note that percentages in the figure may sum to more than 100 because a single jump can be associated with multiple news events.

[12] For more information on this event, we refer to the official Press release and the related FOMC Meeting Statement. Major financial news outlets covered and analyzed this decision: see e.g., 'Bold Fed goes for half-point cut' and 'Bank acts boldly to avert recession risk'.

the day, the ETF and synthetic log prices move closely together, but they diverge at the announcement. As previously noted, if news reached the entire market instantly, was interpreted homogeneously, and trading were continuous, jumps in a group of stocks should occur simultaneously with the ETF index jump, so spreads should be small and random. Instead, the bottom panel shows that the ETF-synthetic index spread temporarily expands at or immediately after the announcement and then contracts again as constituent prices adjust. That is, markets took a few minutes to incorporate the Fed's news into the stock prices of the Dow 30.

Figure 7: When the FOMC statement is released on Sep. 18, 2007, the synthetic index–ETF spread expands and contracts again

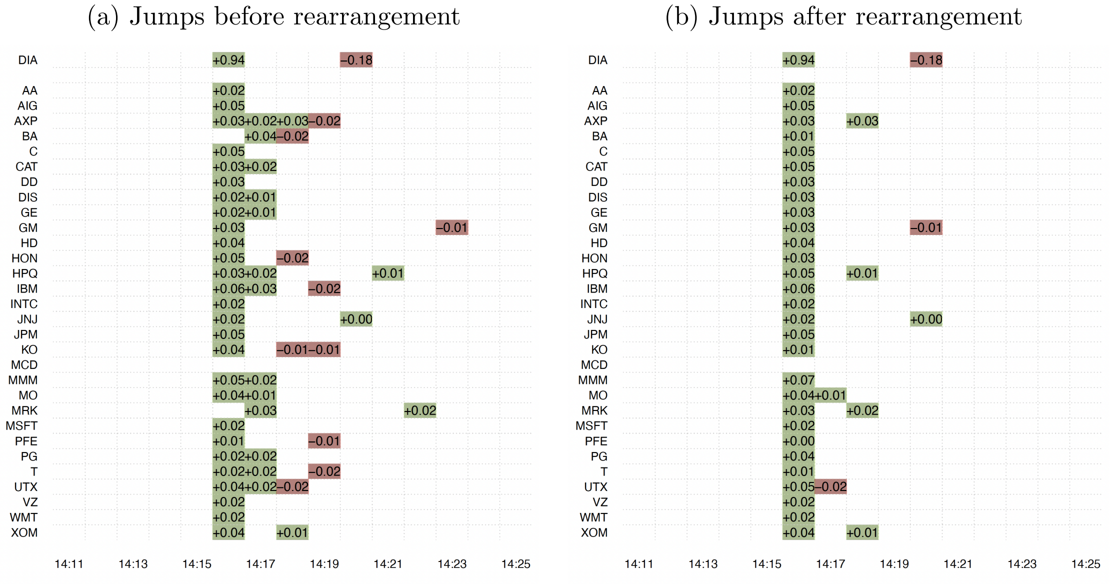


Note: The figure plots the one-minute log prices of the DIA ETF (in gray) and a synthetically constructed Dow 30 index (in black), together with their log return spreads $\delta^r_{i\Delta_n}$ in %. The vertical dashed line marks the release of the FOMC Statement at 2:15 pm. The ETF price jumps shortly after the announcement: A 0.938% jump at 2:16 pm and a −0.184% jump at 2:20 pm are both indicated with red circles and ticks.

The left panel of Figure 8 provides a clearer view of how individual Dow 30 stock prices adjusted to the Fed's announcement in the event window. The figure shows 55 individual stock jumps across the 30 constituents, of which 28 occur simultaneously with the ETF jump, while 27 occur with a lag and are therefore candidates for rearrangement. The right panel shows that jumps are more aligned after (an optimal) rearrangement, illustrating how the procedure corrects for sluggish jumps in the raw data.[13]

[13] Some of the initial jumps (both for the stocks and the ETF) are soon partially reversed by much smaller jumps of the opposite sign a few minutes later. Such a pattern could be related to one of the explanations for negative autocorrelation and/or overshooting in asset returns. Among these explanations, we view order flow dynamics as providing the most plausible explanation for our overshooting jumps.

Figure 8: Stock jumps on Sep. 18, 2007 before and after rearrangement


(a) Jumps before rearrangement

(b) Jumps after rearrangement

Note: Each panel shows the jumps detected in the DIA and Dow 30 constituents around the FOMC Statement of Sep. 18, 2007 at 2:15 pm. Panel (a) shows the jumps in raw returns. Panel (b) shows the same jumps after applying the rearrangement procedure. Positive jumps are shown in green, and negative jumps in red.

For the Sep. 18, 2007 announcement, the jump-event matrix covers an interval from 2:11 pm to 2:25 pm. The jump-event matrix contains $h = 15$ rows, one for each minute in the window, and $q = 56$ columns, where there are 55 jumps in individual stock prices. This produces multiple, very large, constraint matrices. The co-permutation matrix in (20), which contains the main decision variables, is of dimension $h \times (hq)$, i.e., 12,600 elements. Despite this dimensionality, the algorithm is computationally tractable. A standard personal computer can optimally rearrange the jumps for this event in approximately 22 seconds with the Gurobi Optimizer.

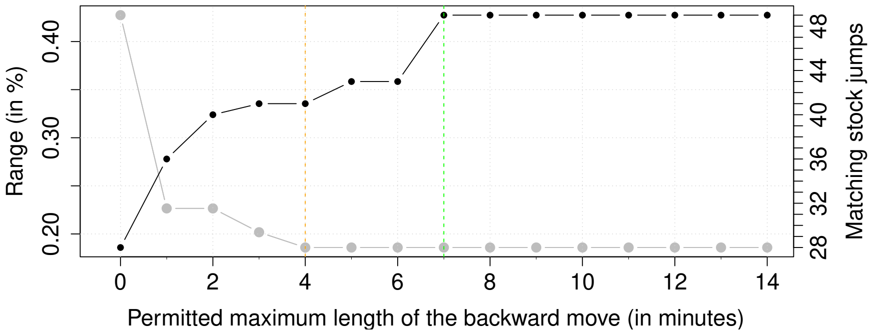
One can look at the improvement in misalignment as a function of the constraint on the size of backward moves. By construction, misalignment cannot increase when one loosens the constraint, i.e., allows larger backward moves. Figure 9 shows the range of the return spreads (left scale, gray) and the number of matched stocks (right scale, black) as a function of the permitted length each individual jump may be moved, as in Equation (30). Permitting each jump to move four minutes backward minimizes the range (orange line); the range drops from 0.427% to 0.186%. Permitting a maximum backward move of 7 periods or 7 minutes (green line) produces the same minimal range for a larger number of matching stocks.[14] For this event, this rearrangement moves 24 out of 27 scattered

That is, news shock pushes traders in the same direction, widening spreads, lowering depth and soon pushing prices too far. To balance their inventories by attracting offsetting trades, dealers mark prices aggressively in the other direction. We readily admit, however, that models of rational inattention or behavioral biases might lead investors to overreact to initial price movements, an overreaction that is soon reversed.

[14] Note that the range does not continue to decrease after the maximum length of backward moves exceeds 4, but the number of matching stock jumps continues to increase until it reaches a maximum for a permitted length of backward move of 7. That is, the solutions at lengths of 4 and 7 produce equivalent

jumps around in the window and matches 21 out of 27 scattered stock jumps with the ETF jump, approximately recovering the common jump in the stocks, as can be seen in the right panel of Figure 8.

Figure 9: Trace plot of return spread and matched stocks (Sep. 18, 2007)



Note: This figure plots the range of the return spreads (left scale, gray) and the number of matching stocks (right scale, black) as a function of the number of the permitted length of the move during the Sep. 18, 2007 FOMC statement. We permit backward moves of length 0 to 14 minutes and solve the RMILP for each of these constraints. An orange line marks the rearrangement with minimal range and a green line marks the rearrangement with minimal range and the maximum number of matching stock jumps.

3.4 Jump rearrangement and dependence among stocks

Recovering common jumps should decrease return misalignment and improve estimates of the daily realized covariance matrix. Following Barndorff-Nielsen and Shephard (2004), the realized covariance matrix is defined as the cross-product of high-frequency stock returns:

$$C = \sum_{i=1}^{\lfloor T/\Delta_n \rfloor} (\Delta_i^n Y)(\Delta_i^n Y)^\top, \tag{31}$$

where returns are calculated from log price increments $\Delta_i^n Y = Y_{i\Delta_n} - Y_{(i-1)\Delta_n}$, and $Y_{i\Delta_n} = (Y_{1,\Delta_n}, \ldots, Y_{p,\Delta_n})^\top$ denotes the p-dimensional vector of stock prices sampled on a regular time grid $\{i\Delta_n : 0 \leq i \leq \lfloor T/\Delta_n \rfloor\}$. As is standard, we set the time span to $T = 1$, and compute 30×30 realized covariances over each trading day, for a sequence of N days. The realized covariance estimator with rearranged returns should better reflect co-movements in the latent, efficient prices than the standard realized covariance matrix (31) with unrearranged returns.

Researchers have proposed several alternative estimators to address related biases in high-frequency covariance estimation. For example, the multivariate realized kernel in Barndorff-Nielsen et al. (2011) and the Cholesky factorization in Boudt et al. (2017), protect against market microstructure noise and address the Epps (1979) effect, that is, the tendency of return correlations in related assets to decline as data frequency increases. In this paper, we show that using rearranged returns in (31) also corrects the underestimation of jump dependence.

ranges, but we choose the solution with more matches.

We now illustrate the effect of the rearrangement on realized jump return covariances, which are one component of total realized return covariances. The realized jump covariance is the cross-product of jump returns:

$$CJ = \sum_{i=1}^{\lfloor T/\Delta_n \rfloor} (\Delta_i^n J)(\Delta_i^n J)^\top, \tag{32}$$

where $\Delta_i^n J$ is a p-dimensional vector of jump returns.

Figure 10 compares pairwise realized jump covariances, i.e., the off-diagonal entries of CJ, for Sep. 18, 2007, under (i) coarser sampling (left panel) and (ii) rearrangement (right panel). The left-hand panel of Figure 10 shows a jump-setting analogue to the classic Epps effect: realized jump covariances at 5 minutes are generally larger than those at 1 minute, resulting in most dots being above the 45 degree line. That is, mistimed jumps at a high-frequency lead to an underestimation of jump dependence. In particular, jump covariances calculated from 1-minute returns (x-axis) are often close to zero, but shift upward when calculated from 5-minute returns (y-axis), as highlighted in red. In other words, aggregating returns to 5-minute intervals produces larger covariances.

The right-hand panel of Figure 10 shows how rearranging returns affects realized covariances. If rearrangement did not actually move returns, all the dots would lie on the 45 degree line in the right-hand panel of Figure 10. The dots in the right-hand panel, however, lie mostly above the 45 degree line, indicating that rearrangement mostly increases the estimated jump covariances. In particular, jump covariances that are near zero in the raw 1-minute data (red dots) become substantially larger after rearrangement, recovering cross-sectional dependence that would otherwise only be discovered through temporal aggregation.

Mistimed jumps can degrade estimates of parameters crucial to portfolio decisions, including attenuating the estimates of jump betas. Li, Todorov, and Tauchen (2017) propose to estimate jump-betas as the ratio of the realized jump covariation between stock and the ETF returns to the realized jump variation:

$$\widehat{\beta} = \frac{\sum_{i=1}^{\lfloor T/\Delta_n \rfloor} \Delta_i^n Y^d \Delta_i^n Z^d}{\sum_{i=1}^{\lfloor T/\Delta_n \rfloor} (\Delta_i^n Z^d)^2}, \tag{33}$$

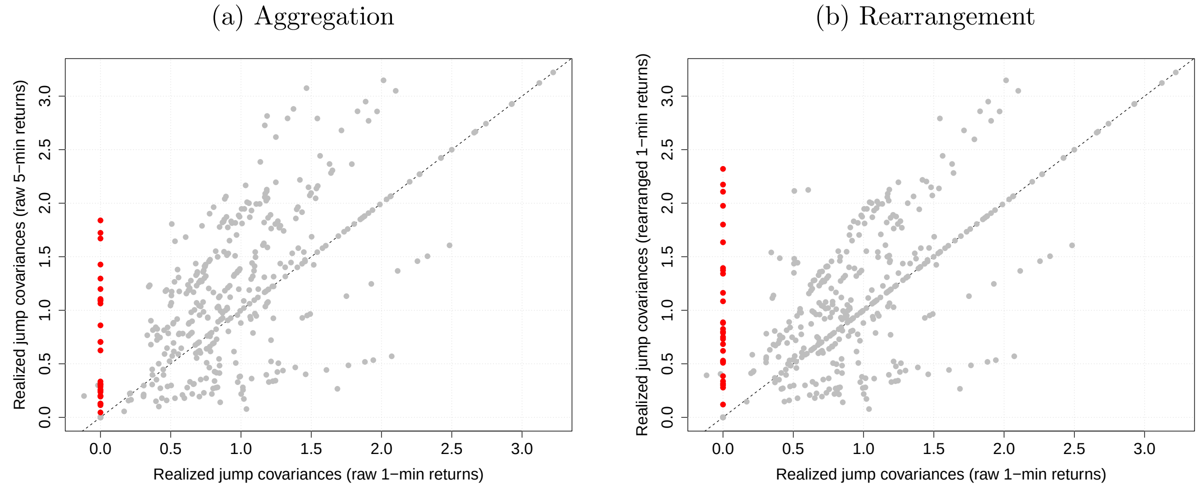
where $Y_{i\Delta_n}^d = (Y_{1,\Delta_n}^d, \ldots, Y_{p,\Delta_n}^d)^\top$ denotes the jump component of the log price process in (2) for the p stocks, sampled on a regular time grid $\{i\Delta_n : 0 \leq i \leq \lfloor T/\Delta_n \rfloor\}$ over T days, and Z^d denotes the jump component of the ETF price process. Note that a feasible estimate of $\widehat{\beta}$ uses estimated jump returns, as in (32).

Mistimed jumps bias the estimated jump betas toward zero by creating discrepancies between the observed jump process Y^d and the latent efficient jump process X^d. Our rearrangement procedure mitigates this bias.

Figure 11 illustrates how rearrangement and sampling frequency affect the magnitude of realized jump betas. We restrict the sample to days of rearrangement of stock returns. The left panel of Figure 11 compares average daily realized jump betas computed on raw 1-minute jump returns and rearranged 1-minute jump returns. For 38 out of 43 stocks, rearrangement increases the average realized jump beta, with the cross-sectional median increasing from 0.67 to 1.01. In one case, a negative beta turns positive, eliminating what would otherwise appear to be a spurious hedging opportunity.

The most extreme revision occurs for Alcoa, whose realized jump beta rises from 0 to 5. This is driven by a single rearrangement on February 10, 2009, when a jump in

Figure 10: Rearrangement recovers cross-sectional dependence without temporal aggregation

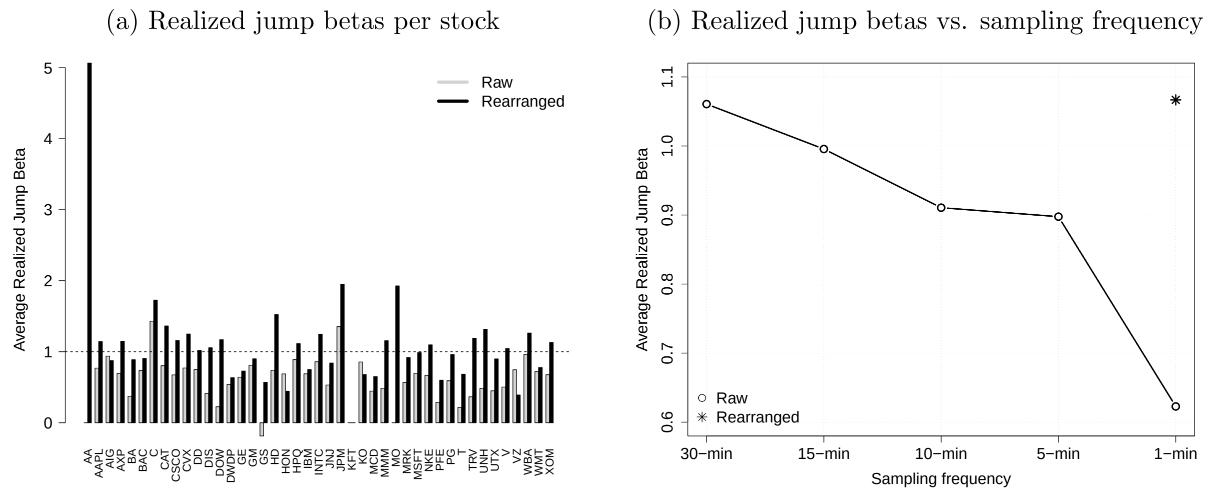


(a) Aggregation

(b) Rearrangement

Note: The scatterplots compare $30 \times 29/2 = 435$ off-diagonal entries of the realized jump covariance matrix of Dow 30 stocks on September 18, 2007. Panel (a) plots realized covariances from raw 1-minute jump returns (x-axis) and against those from 5-minute returns (y-axis). The dotted line corresponds to the 45-degree line. Panel (b) plots the 1-minute raw realized jump covariances (x-axis) against 1-minute jump covariances computed after applying the rearrangement procedure (y-axis). Entries that shift from zero to positive values are highlighted in red.

Figure 11: Effect of rearrangement on realized jump beta estimates using 1-minute returns



(a) Realized jump betas per stock

(b) Realized jump betas vs. sampling frequency

Note: Panel (a) shows average realized jump betas over the full sample for each Dow Jones constituent, using raw versus rearranged 1-minute jump returns. Panel (b) shows average realized jump betas over the full sample at different sampling frequencies.

Alcoa at 11:12 am was shifted backward to coincide with an ETF jump at 11:07 am.[15]

[15] For more information on this event, we refer to the official Press release. The full speech of the U.S. Treasury Secretary and video can be found here. The speech was broadly covered in the FT:

Three pieces of news drove the ETF and Alcoa jumps that morning. At 11:00 am, the Federal Reserve announced that it was prepared to substantially expand the Term Asset-Backed Securities Loan Facility (TALF) and broaden the eligible collateral. Around the same time, U.S. Treasury Secretary Timothy Geithner unveiled the administration's much-anticipated Financial Stability Plan, which disappointed markets with a lack of detail. Presumably as a result of negative interpretations of these announcements, the ETF price jumped at 11:07. At 11:08, the S&P announced that it had cut Alcoa's credit rating in view of deteriorating business conditions. The combination of these news about the economy and Alcoa's credit rating caused Alcoa's stock price to jump downwards at 11:12. Excluding this observation, the median realized jump beta across all stocks still rises markedly, from 0.68 to 1.00.

The right panel of Figure 11 plots the average realized jump betas over all stocks for raw returns by sampling frequency, as well as the average realized jump beta for rearranged returns at the 1-minute frequency. Consistent with the hypothesis that mistimed jumps attenuate realized jump betas, rearrangement produces substantially higher jump-beta estimates. Rearranged returns from 1-minute data produce realized jump betas comparable to those produced from 30-minute data.

Having shown that rearrangement generally increases realized jump beta estimates, we now investigate whether these new estimates improve return predictions. Building on Ding et al. (2024), who document asset jumps predict subsequent performance, we compare the forecasting accuracy of realized jump betas computed from rearranged versus raw 1-minute returns. Our sample comprises 238 observations across 83 days on which stock returns were rearranged (see column #RA in Table 1). We estimate the following pooled predictive regression:

$$r_{i,t+1} = \gamma_0 + \gamma_1 \widehat{\beta}_{i,t}^{Rearr} + \gamma_2 \widehat{\beta}_{i,t}^{Raw} + \gamma_3 r_{i,t} + \gamma_4 RTSRV_{i,t} + \varepsilon_{i,t+1}, \tag{34}$$

where $r_{i,t+1}$ denotes the next-day return of stock i, and $\widehat{\beta}_{i,t}^{Rearr}$ and $\widehat{\beta}_{i,t}^{Raw}$ are the realized jump betas estimated from rearranged and raw returns, respectively. Two firm-level controls are included: the contemporaneous daily return $r_{i,t}$ and the noise- and jump-robust measure of daily integrated variance $RTSRV_{i,t}$ of Boudt and Zhang (2015).

Table 2 reports ordinary least squares estimates of (34). Model 3 represents the full specification, while Models 1 and 2 impose $\gamma_1 = 0$ (raw realized jump beta only) and $\gamma_2 = 0$ (rearranged jump berealized ta only), respectively. The adjusted R^2 of Model 3 (11.8%) indicates meaningful predictability of next-day returns following a rearrangement. Both $\widehat{\gamma}_1$ and $\widehat{\gamma}_2$ are negative, but only the coefficient on the rearranged beta is statistically significant at the 5%, suggesting that the relevant information for predicting subsequent returns is embedded in the rearranged component. Moreover, Model 2 explains more variation in returns than Model 1 (11.9% vs. 10.3%). $\widehat{\gamma}_3$ is also negative and significant at conventional significance levels, indicating a sharp market correction in prices following a rearrangement day.

We evaluate the predictive power of realized jump betas with an out-of-sample forecasting exercise. The estimation window spans all rearrangements up to 2009-11-23, leaving a testing period from 2010-01-05 to 2020-01-08 with 164 observations on 66 rearrangement days. We estimate the three models on expanding windows to forecast next-day returns.

see e.g., Overview: Toxic asset scheme triggers sell-off, US bank bail-out, Wall street falls on Geithner announcement, Wall street: not impressed. More information on the rating cut of Alcoa can be found here.

Model 2 achieves the lowest out-of-sample, mean-squared forecast error and remains as the only superior model within the Model Confidence Set (MCS) of Hansen et al. (2011) at a nominal level of 10%.[16] This result shows that realized jump betas calculated on rearranged returns have greater predictive power for daily returns than jump betas calculated on raw returns.

We then use predicted returns from each model to form simple trading rules. The long-only strategy takes a position when the forecasted return is positive, whereas the long–short strategy goes long (short) when the forecasted return is positive (negative), with a one-day holding period. The second panel shows the model based on rearranged realized jump betas (Model 2) substantially outperforms its raw-return counterpart (Model 1). The former yields cumulative returns of 42.7% and 53.0% for the long-only and long–short strategies, respectively, compared with 26.3% and 20.2% for Model 1. The corresponding Sharpe ratios are 0.214 vs. 0.079, and the difference is statistically significant at the 5% level according to the test of Ledoit and Wolf (2008) implemented using a studentized circular bootstrap with 499 resamples and two-sided p-values computed as in Barras et al. (2010, Appendix A.3).

These results demonstrate that rearranging high-frequency returns before estimating realized jump betas not only improves statistical forecast accuracy, but also translates into economically meaningful gains in portfolio performance. Correcting for jump-timing misalignment across assets uncovers systematic co-movement that enhances both our measurement of market jumps and our understanding of how information diffuses through asset prices.

4 Concluding remarks

During news events, stock price jumps often occur at close but distinct points in time on a fine sampling grid. Such asynchronous jumps bias covariance estimates and degrade portfolio strategies. To optimally rearrange asynchronous jumps in time and better recover the common jump, we introduce tractable, new tools that are based on the combinatorial methods of Puccetti and Rüschendorf (2012) and Embrechts et al. (2013). We carefully detail these novel procedures, provide toy examples to assist intuition and discuss a real-world example of delayed jumps.

We use Dow 30 prices to link delayed jumps to scheduled macroeconomic announcements. Rearranged jumps are particularly likely to follow Federal Reserve communications, which seem to be the most important source of jumps. Using returns of varying frequencies, we illustrate how jump rearrangement with 1-minute returns improves realized jump beta estimates in a manner similar to time aggregation but without the loss of precision. Realized jump betas from rearranged returns outperform jump betas from raw data in out-of-sample return forecasting exercises. Trading rules constructed from rearranged returns also outperform the same rules using raw returns.

Recovering the common jump on a fine sampling grid is also likely to improve other asset allocation and risk management decisions, such as estimating the distribution of jump sizes (see *e.g.*, Boudt et al., 2011a), estimating jump dependence (see *e.g.*, Li, Todorov,

[16] The MCS procedure of Hansen et al. (2011) identifies the subset of models that are statistically indistinguishable in terms of predictive ability at a given confidence level. We use the maximum statistic, and compute p-values via an i.i.d. bootstrap with 10,000 resamples. Models are iteratively eliminated until the null hypothesis of equal predictive ability cannot be rejected.

Table 2: In-sample regression results and out-of-sample statistical/financial performance

	Model 1	Model 2	Model 3
In-sample		2007-02-27 to 2020-01-08	
$\gamma_0 \rightarrow Constant$	0.004	0.006	0.006
	(2.46)	(3.23)	(3.29)
$\gamma_1 \rightarrow \widehat{\beta}_{i,t}^{Rearr}$	–	-0.004	-0.003
		(-3.05)	(-2.19)
$\gamma_2 \rightarrow \widehat{\beta}_{i,t}^{Raw}$	-0.003	–	-0.001
	(-2.21)		(-0.72)
$\gamma_3 \rightarrow r_{i,t}$	-0.551	-0.530	-0.538
	(-5.00)	(-4.88)	(-4.92)
$\gamma_4 \rightarrow RTSRV_{i,t}$	-2.321	-2.472	-2.390
	(-1.12)	(-1.21)	(-1.16)
Adjusted R^2	0.103	0.119	0.118
Out-of-sample		2010-01-05 to 2020-01-08	
Statistical performance			
MSFE$\times 10^3$	0.321	0.263	0.345
MCS [p-value]	0.099	1.000	0.099
Financial performance			
		Long-only	
Cumulative return	0.280	0.426	0.333
Standard deviation	0.013	0.014	0.013
Sharpe ratio	0.134	0.191	0.159
Sharpe ratio test [p-value]	[0.020]		
		Long-short	
Cumulative return	0.194	0.485	0.300
Standard deviation	0.015	0.015	0.015
Sharpe ratio	0.078	0.200	0.122
Sharpe ratio test [p-value]	[0.032]		

Note: The first panel reports the estimation results of the model $r_{i,t+1} = \gamma_0 + \gamma_1 \widehat{\beta}_{i,t}^{Rearr} + \gamma_2 \widehat{\beta}_{i,t}^{Raw} + \gamma_3 r_{i,t} + \gamma_4 RTSRV_{i,t} + \varepsilon_{i,t+1}$ over the full sample. γ_1 (resp. γ_2) is restricted to zero in Model 1 (resp. Model 2). t-statistics are in parentheses. The second panel reports the results of an out-of-sample forecasting exercise. The first block summarizes the statistical performance of each model. The row ‘$MSFE \times 10^3$’ reports the mean squared forecast error (multiplied by 10^3) and ‘MCS [p-value]’ reports the p-value of the Model Confidence Set of Hansen et al. (2011). The second block reports financial performance. Under ‘long-only’, the strategy takes a long position when the predicted return is positive. Under ‘long–short ’, it goes long (short) when the predicted return is positive (negative). The holding period is one day. Cumulative return, standard deviation, and Sharpe ratio summarize each model’s trading results. The row ‘Sharpe ratio test [p-value]’ reports the two-sided p-value from the test of Ledoit and Wolf (2008) for equality of Sharpe ratios between Models 1 and 2.

Tauchen, and Chen, 2017; Li, Todorov, and Tauchen, 2017; Li, Todorov, Tauchen, and Lin, 2019), or forecasting realized measures (see *e.g.*, Andersen et al., 2007; Bollerslev et al., 2022). A more thorough analysis must, however, await future work.

While our narrative has emphasized sluggish impoundment of systemic news – whereby the ETF reacts promptly and individual constituents follow with delays – an inverse mechanism is also possible. Idiosyncratic news may trigger a jump in a large-weight stock that immediately moves the index. For price-weighted indices such as the DJIA, this channel can be important. Our framework accommodates such cases by treating them as special episodes in which no rearrangement is required, but a natural extension would be to relax the backward-only restriction and allow for the possibility that highly liquid, large-weight constituents occasionally lead the ETF. Such "leading stocks" are rare, but modeling them directly could be valuable for applications to broader indices, such as the S&P 500.

References

Aït-Sahalia, Y. (2004). Disentangling diffusion from jumps. Journal of Financial Economics 74(3), 487–528.

Aït-Sahalia, Y. and J. Jacod (2014). High-frequency financial econometrics. Princeton University Press.

Andersen, T. G., T. Bollerslev, and F. X. Diebold (2007). Roughing it up: Including jump components in the measurement, modeling, and forecasting of return volatility. Review of Economics and Statistics 89(4), 701–720.

Andersen, T. G., T. Bollerslev, F. X. Diebold, and C. Vega (2003). Micro effects of macro announcements: Real-time price discovery in foreign exchange. American Economic Review 93(1), 38–62.

Aït-Sahalia, Y., J. Cacho-Diaz, and T. R. Hurd (2009). Portfolio choice with jumps: A closed-form solution. The Annals of Applied Probability 19(2), 556–584.

Bandi, F. and R. Renò (2016). Price and volatility co-jumps. Journal of Financial Economics 119(1), 107–146.

Bandi, F. M., D. Pirino, and R. Renò (2017). Excess idle time. Econometrica 85(6), 1793–1846.

Barndorff-Nielsen, O. E., P. R. Hansen, A. Lunde, and N. Shephard (2009). Realized kernels in practice: Trades and quotes. Econometrics Journal 12(3), C1–C32.

Barndorff-Nielsen, O. E., P. R. Hansen, A. Lunde, and N. Shephard (2011). Multivariate realised kernels: consistent positive semi-definite estimators of the covariation of equity prices with noise and non-synchronous trading. Journal of Econometrics 162(2), 149–169.

Barndorff-Nielsen, O. E. and N. Shephard (2004). Econometric analysis of realized covariation: High frequency based covariance, regression, and correlation in financial economics. Econometrica 72(3), 885–925.

Barras, L., O. Scaillet, and R. Wermers (2010). False discoveries in mutual fund performance: Measuring luck in estimated alphas. The journal of finance 65(1), 179–216.

Bibinger, M. and L. Winkelmann (2015). Econometrics of co-jumps in high-frequency data with noise. Journal of Econometrics 184(2), 361–378.

Bollerslev, T., T. H. Law, and G. Tauchen (2008). Risk, jumps, and diversification. Journal of Econometrics 144(1), 234–256.

Bollerslev, T., A. J. Patton, and R. Quaedvlieg (2022). Realized semibetas: Disentangling "good" and "bad" downside risks. Journal of Financial Economics 144(1), 227–246.

Bondarenko, O. and C. Bernard (2024). Option-implied dependence and correlation risk premium. Journal of Financial and Quantitative Analysis, 3139 – 3189.

Boudt, K., C. Croux, and S. Laurent (2011a). Outlyingness weighted covariation. Journal of Financial Econometrics 9(4), 657–684.

Boudt, K., C. Croux, and S. Laurent (2011b). Robust estimation of intraweek periodicity in volatility and jump detection. Journal of Empirical Finance 18(2), 353–367.

Boudt, K., E. Jakobsons, and S. Vanduffel (2018). Block rearranging elements within matrix columns to minimize the variability of the row sums. 4OR 16(1), 31–50.

Boudt, K., S. Laurent, A. Lunde, R. Quaedvlieg, and O. Sauri (2017). Positive semidefinite integrated covariance estimation, factorizations and asynchronicity. Journal of Econometrics 196(2), 347–367.

Boudt, K. and J. Zhang (2015). Jump robust two time scale covariance estimation and realized volatility budgets. Quantitative Finance 15(6), 1041–1054.

Caporin, M., A. Kolokolov, and R. Renò (2017). Systemic co-jumps. Journal of Financial Economics 126(3), 563–591.

Christensen, K., R. C. Oomen, and M. Podolskij (2014). Fact or friction: Jumps at ultra high frequency. Journal of Financial Economics 114(3), 576–599.

Christensen, K., A. Timmermann, and B. Veliyev (2023). Warp speed price moves: Jumps after earnings announcements. Available at SSRN 4422376.

Ding, Y., Y. Li, G. Liu, and X. Zheng (2024). Stock co-jump networks. Journal of Econometrics 239(2), 105420.

Embrechts, P., G. Puccetti, and L. Rüschendorf (2013). Model uncertainty and VaR aggregation. Journal of Banking & Finance 37(8), 2750–2764.

Epps, T. W. (1979). Comovements in stock prices in the very short run. Journal of the American Statistical Association 74(366a), 291–298.

Gilder, D., M. B. Shackleton, and S. J. Taylor (2014). Cojumps in stock prices: Empirical evidence. Journal of Banking Finance 40, 443–459.

Gnabo, J.-Y., L. Hvozdyk, and J. Lahaye (2014). System-wide tail comovements: A bootstrap test for cojump identification on the sp 500, us bonds and currencies. Journal of International Money and Finance 48, 147–174.

Hansen, P. R., A. Lunde, and J. M. Nason (2011). The model confidence set. Econometrica 79(2), 453–497.

Jacod, J., V. Todorov, et al. (2009). Testing for common arrivals of jumps for discretely observed multidimensional processes. The Annals of Statistics 37(4), 1792–1838.

Jondeau, E., S.-H. Poon, and M. Rockinger (2007). Financial modeling under non-Gaussian distributions. Springer Science & Business Media.

Lahaye, J., S. Laurent, and C. J. Neely (2011). Jumps, cojumps and macro announcements. Journal of Applied Econometrics 26(6), 893–921.

Ledoit, O. and M. Wolf (2008). Robust performance hypothesis testing with the sharpe ratio. Journal of Empirical Finance 15(5), 850–859.

Lee, S. S. and P. A. Mykland (2008). Jumps in financial markets: A new nonparametric test and jump dynamics. Review of Financial Studies 21(6), 2535–2563.

Lee, S. S. and P. A. Mykland (2012). Jumps in equilibrium prices and market microstructure noise. Journal of Econometrics 168(2), 396–406.

Li, J., V. Todorov, and G. Tauchen (2017). Jump regressions. Econometrica 85(1), 173–195.

Li, J., V. Todorov, G. Tauchen, and R. Chen (2017). Mixed-scale jump regressions with bootstrap inference. Journal of Econometrics 201(2), 417–432.

Li, J., V. Todorov, G. Tauchen, and H. Lin (2019). Rank tests at jump events. Journal of Business & Economic Statistics 37(2), 312–321.

Neely, C. J., A. Cole, N. Bouamara, K. Boudt, and S. Laurent (Sept. 29, 2025). What causes 'jumps' in stock prices? Technical report, St. Louis Fed On the Economy.

Puccetti, G. and L. Rüschendorf (2012). Computation of sharp bounds on the distribution of a function of dependent risks. Journal of Computational and Applied Mathematics 236(7), 1833–1840.