Supplementary Appendix: Technical Note and Simulation Protocol

Section 1: Methodological Rationale and Simulation Architecture

In standard pharmaceutical randomized controlled trials (RCTs), the intervention node is stable, standardized, and perfectly exchangeable across centers. Statistical frameworks applied to these trials rely heavily on the assumption of independent and identically distributed (i.i.d.) observations. In surgical science, however, the intervention is intrinsically operator-dependent, temporally evolving, technically heterogeneous, and hierarchically clustered within surgeons and institutions.

When multi-surgeon or multi-center surgical RCT data are analyzed using unadjusted conventional statistical models (such as the naïve Cox proportional hazards model), the fundamental assumption of independence is violated. This gives rise to **pseudo-replication**. Furthermore, in real-world surgical contexts, a critical unobserved confounding mechanism often manifests: certain high-volume or specific-tier centers/surgeons are structurally predisposed to adopt or be assigned to a novel surgical approach (e.g., minimally invasive surgery). When the allocation probability correlates even mildly with unobserved cluster-level institutional or provider traits (modeled via a logistic propensity mechanism), ignoring this hierarchical clustering yields catastrophic statistical consequences.

This Monte Carlo simulation explicitly demonstrates the methodological vulnerabilities of applying conventional drug-trial statistical frameworks to clustered surgical data under a strict biological null hypothesis—where the true hazard ratio (HR) between the innovative intervention and the standard arm is exactly 1.0 (zero treatment effect).

We contrast two distinct analytical frameworks across 500 independent experimental iterations:

1. **The Naïve Cox Model:** A standard Cox proportional hazards model that treats all patient observations as independent, completely ignoring surgeon-level clustering and structural covariance.
2. **The Shared Frailty Cox Model:** A random-effects survival model that incorporates a Gaussian frailty term ($\tau^2=1.0$) to explicitly capture and adjust for surgeon-specific baseline risks and clustering dependencies.

The simulation illustrates that ignoring surgeon-level correlation artificially shrinks the standard error (SE) estimates, resulting in a dramatic inflation of the **Type I Error Rate (False-Positive Rate)** far beyond the nominal 5% significance threshold. This highlights how spurious "statistically significant" survival differences can easily emerge in surgical trials purely due to unmodeled provider-level heterogeneity.

Section 2: Annotated R Implementation Code

The complete, fully commented R code below is optimized for academic transparency and direct execution within the R environment, generating a publication-grade four-panel composite figure (Panels A–D) utilizing the patchwork framework.

R

```
################################################################################
# Figure 2. Monte Carlo Simulation: Methodological Impacts of Surgeon Clustering
# Protocol: Parameter-Optimized Shared Frailty Cox vs. Naïve Cox Models
# Designed for Publication-Grade Appendix
################################################################################

# Load core survival analysis and data visualization packages
library(tidyverse)
library(survival)
library(patchwork)

# ==============================================================================
# 1. PARAMETER DEFINITION & SEED INITIALIZATION
# ==============================================================================
set.seed(2026)        # Set fixed random seed for exact mathematical reproducibility
n_sims     <- 500     # Number of Monte Carlo iterations (optimal for non-linear survival)
n_surgeons <- 30      # Total number of surgical clusters (Surgeons)
n_patients <- 20      # Cluster size (Number of patients per surgeon)
N          <- n_surgeons * n_patients # Total sample size across trial (N = 600)

# Survival analysis architectural parameters
true_beta  <- 0.0     # True biological treatment effect (Beta = 0; True HR = 1.0 for Type I error)
tau_sq     <- 1.0     # Intrinsic surgeon-level random effect variance (Between-surgeon variance)
lambda_0   <- 0.1     # Baseline hazard rate for Exponential survival distribution

# ==============================================================================
# 2. DATA CONTAINER INITIALIZATION
# ==============================================================================
sim_results <- tibble(
  sim_id      = 1:n_sims,
  naive_hr    = NA_real_, naive_se  = NA_real_, naive_p  = NA_real_,
  frailty_hr  = NA_real_, frailty_se = NA_real_, frailty_p = NA_real_
)

# ==============================================================================
# 3. MONTE CARLO SIMULATION LOOP
# ==============================================================================
message(">>> Running Monte Carlo survival simulations, please wait...")

for(i in 1:n_sims) {

  # Draw independent surgeon-specific random effects (frailties) for this iteration
  surgeon_effects <- rnorm(n_surgeons, mean = 0, sd = sqrt(tau_sq))

  # Construct the hierarchical patient-level survival clinical dataset
  sim_data <- tibble(
    surgeon_id = rep(1:n_surgeons, each = n_patients),
    w_i        = rep(surgeon_effects, each = n_patients)
  ) %>%
```

```
  # 【CRITICAL MECHANISM OPTIMIZATION】: Model realistic surgical trial scenarios where
  # the assignment probability of a new technique is structurally correlated with
  # unobserved cluster traits (e.g., surgeon preference, proficiency, or center volume).
  # This breaks the independence assumption and correctly simulates Type I error inflation.
  group_by(surgeon_id) %>%
  mutate(
    # Map surgeon random effect to a propensity score via a logistic link function
    prob_b      = 1 / (1 + exp(-0.7 * w_i)),
    prob_assign = pmax(0.2, pmin(0.8, prob_b)), # Bound probabilities to ensure events in both arms
    treatment   = rbinom(n(), 1, prob = prob_assign[1])
  ) %>%
  ungroup() %>%
  mutate(
    # Generate Exponential survival times using the inverse cumulative hazard transform
    # Formula: Time = -log(U) / (lambda_0 * exp(beta * X + w_i))
    U = runif(N),
    survival_time = -log(U) / (lambda_0 * exp(true_beta * treatment + w_i)),

    # Introduce independent uniform right-censoring mechanisms
    censor_time   = runif(N, min = 5, max = 35),
    status        = if_else(survival_time <= censor_time, 1, 0),
    time          = pmin(survival_time, censor_time)
  )

 # A. Naïve Cox Model: Completely ignores surgeon-level hierarchical clustering
 fit_naive <- coxph(Surv(time, status) ~ treatment, data = sim_data)
 sum_naive <- summary(fit_naive)$coefficients

 sim_results$naive_hr[i] <- sum_naive["treatment", "exp(coef)"]
 sim_results$naive_se[i] <- sum_naive["treatment", "se(coef)"]
 sim_results$naive_p[i]  <- sum_naive["treatment", "Pr(>|z|)"]

 # B. Shared Frailty Cox Model: Incorporates surgeon cluster into a Gaussian frailty term
 fit_frail <- coxph(Surv(time, status) ~ treatment +
                    frailty(surgeon_id, distribution = "gaussian"), data = sim_data)
 sum_frail <- summary(fit_frail)

 sim_results$frailty_hr[i] <- exp(fit_frail$coefficients["treatment"])
 # Extract robust standard error that correctly accounts for intra-cluster correlation
 sim_results$frailty_se[i] <- sum_frail$coefficients["treatment", "se(coef)"]
 sim_results$frailty_p[i]  <- sum_frail$coefficients["treatment", "p"]
}

message(">>> Simulation complete. Rendering publication-grade four-panel visualization...")

# ==============================================================================
# 4. CAPTURING REPRESENTATIVE SURGEON EFFECTS (For Panel A Static Visualization)
# ==============================================================================
df_panelA <- tibble(
  surgeon = factor(1:n_surgeons),
  effect  = surgeon_effects
) %>%
  arrange(effect) %>%
  mutate(surgeon = fct_inorder(surgeon)) # Sort sequentially for rigorous visual aesthetic

# ==============================================================================
# 5. JOURNAL-GRADE RASTERING AND RENDERING (PANELS A - D)
# ==============================================================================
font_family <- "sans"
color_dark  <- "#222222"
```

```
color_blue  <- "#377EB8" # Corrected robust Frailty Models (Blue)
color_red   <- "#E41A1C" # Flawed unadjusted Naïve Models (Red)

# ---- Panel A. Simulated Surgeon-Specific Random Effects ----
pA <- ggplot(df_panelA, aes(x = surgeon, y = effect)) +
  geom_segment(aes(xend = surgeon, yend = 0), color = "grey70", linewidth = 0.5) +
  geom_point(color = color_dark, size = 1.8) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "grey40") +
  labs(
    title = "Panel A. Simulated Surgeon-Specific Random Effects",
    x = "Surgeon ID (Ordered by operational baseline risk)",
    y = "Log Hazard Ratio Contribution (w_i)"
  ) +
  theme_minimal(base_family = font_family, base_size = 11) +
  theme(axis.text.x = element_blank(), panel.grid.major.x = element_blank())

# Define uniform horizontal axis limits to enable direct empirical density comparison
hr_lims <- c(0.4, 2.2)

# ---- Panel B. Naïve Cox Models (Ignoring Clustering) ----
pB <- ggplot(sim_results, aes(x = naive_hr)) +
  geom_density(fill = color_red, alpha = 0.6, color = "white") +
  geom_vline(xintercept = 1.0, linetype = "longdash", color = "grey30", linewidth = 0.7) +
  xlim(hr_lims) +
  labs(
    title = "Panel B. Naïve Cox Models (Ignoring Surgeon-Level Clustering)",
    x = "Estimated Hazard Ratio (HR)",
    y = "Probability Density"
  ) +
  theme_minimal(base_family = font_family, base_size = 11)

# ---- Panel C. Frailty Cox Models (Accounting for Clustering) ----
pC <- ggplot(sim_results, aes(x = frailty_hr)) +
  geom_density(fill = color_blue, alpha = 0.6, color = "white") +
  geom_vline(xintercept = 1.0, linetype = "longdash", color = "grey30", linewidth = 0.7) +
  xlim(hr_lims) +
  labs(
    title = "Panel C. Frailty Cox Models (Accounting for Surgeon-Level Clustering)",
    x = "Estimated Hazard Ratio (HR)",
    y = "Probability Density"
  ) +
  theme_minimal(base_family = font_family, base_size = 11)

# ---- Panel D. Inflation of False-Positive Findings ----
df_type1 <- tibble(
  Model = c("Frailty Cox\n(Clustering-Adjusted)", "Naïve Cox\n(Unadjusted)"),
  Rate  = c(mean(sim_results$frailty_p < 0.05), mean(sim_results$naive_p < 0.05))
)

pD <- ggplot(df_type1, aes(x = fct_rev(Model), y = Rate, fill = Model)) +
  geom_bar(stat = "identity", width = 0.45, alpha = 0.85, show.legend = FALSE) +
  geom_hline(yintercept = 0.05, linetype = "dotted", color = "red", linewidth = 0.9) +
  geom_text(aes(label = scales::percent(Rate, accuracy = 0.1)),
            vjust = -0.5, fontface = "bold", size = 4, family = font_family) +
  scale_fill_manual(values = c(color_blue, color_red)) +
  scale_y_continuous(labels = scales::percent_format(accuracy = 1),
                     limits = c(0, max(df_type1$Rate) + 0.08)) +
  labs(
    title = "Panel D. Inflation of False-Positive Findings",
    x = "Statistical Modeling Approach",
```

```
    y = "Empirical Type I Error Rate (Alpha = 5%)"
  ) +
  theme_minimal(base_family = font_family, base_size = 11) +
  theme(panel.grid.major.x = element_blank())

# ==============================================================================
# 6. COMPOSITE INTEGRATION VIA PATCHWORK ARCHITECTURE
# ==============================================================================
combined_appendix_figure <- (pA + pB) / (pC + pD) +
  plot_annotation(
    title = "Methodological Impacts of Surgeon-Level Heterogeneity in Surgical RCTs",
    subtitle = paste0("Monte Carlo Simulation under True Null Hypothesis (500 iterations | 30 surgeons | N = 600
patients | True Biological HR = 1.0)"),
    theme = theme(
      plot.title    = element_text(face = "bold", size = 14, hjust = 0.5, color = color_dark),
      plot.subtitle = element_text(size = 10.5, hjust = 0.5, color = "grey35", face = "italic")
    )
  )

# Render the multi-panel composite figure onto the active graphical device
combined_appendix_figure
```