Supplemental Information

AI-guided Autonomous Workflows Accelerate Kinetic Analysis of Polymer Depolymerization

Nicholas J. Galan,[1] Britney Baez,[1] Ju-An Kim,[1] Beibei Yao,[1] Nduka Ogbonna,[1] Isaiah Dishner,[1] Jared Bowman,[1] Jesse McGaha,[2] Ankit Shrivastava,[2] Stephen DeWitt,[2] Sheng Dai[1], Craig Bridges,[1] Jeffrey C. Foster,[1] Tomonori Saito[1]

[1]Chemical Sciences Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831, United States

[2]Computing and Computational Sciences Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831, United States

Supplemental Discussion

SD1, Depolymerization System Design: To develop our automated deconstruction workflow, we investigated poly(bisphenol A carbonate) (PC) as a model polymer for depolymerization due to its abundance in post-consumer waste streams and solubility in common organic solvents (e.g., $CHCl_3$, THF, DCM, etc.), enabling homogeneous reaction monitoring, which is critical for accurate kinetic profiling. Homogenous PC depolymerization reactions were carried out via glycolysis using the sterically hindered organic superbase 7-methyl-1,5,7-triazabicylco(4.4.0)dec-5-ene (mTBD) to facilitate reasonable reaction times compared to its unsubstituted counterpart.[1] Acetic acid (AcOH) was found to be an effective quenching agent, significantly impeding the depolymerization for up to 24 hours. We selected the ChemSpeed Swing XL system as our autonomous assistant due to its versatile capabilities such as precision liquid transfer, modularity in workflow design, and capacity to execute numerous reactions in parallel, all of which are amenable for in-depth depolymerization kinetic experiments.

SD2, Selection of Kinetic Data Included in Manuscript: Over the course of this study, a total of 26 automated kinetic runs were conducted, producing 21 samples from each run. In total, 546 individual samples were generated and analyzed. Because many of these kinetic runs were used for process optimization or to aid in the development of various workflow components (e.g., batch processing) and were redundant with respect to initial reaction conditions, these runs are excluded from the subset of the data included in the manuscript (7 automated runs and 147 samples + 3 additional AI-guided runs and 24 samples).

SD3, ORNL INTERSECT Overview: An INTERSECT service was created for the lab for experimentation. The lab service was configured to output data to intermediate storage, report relevant outputs, and receive control inputs. The host file system is monitored for the creation of relevant metadata. On observation of relevant metadata, that metadata is uploaded to intermediate storage hosted in the ORNL Research Cloud. The intermediate storage is monitored for the upload of relevant analysis data. On observation of relevant analysis data, the lab service emits an event to notify other systems of the upload. On receiving a control message, the service writes a control file to the host system that triggers a new experiment operation. An INTERSECT client was created to await events from the lab service, configure and provide data to the learning service, and provide returned commands to the lab service. On observation of a lab event, the client retrieves data from intermediate storage and loads it for operations. Observed data is configured and sent to the learning service which responds with the next configuration for analysis. On observation of analysis configuration, the client creates a control string that is sent to the lab service for configuration.

SD4, Kinetic Model Considerations: The Stagg relationship assumes pseudo-first order kinetics with respect to the concentration of bonds in the polymer sample and a Kuhn distribution of polymer chains (i.e., $Đ = 2$) across the duration of the depolymerization reaction and allows interrogation of depolymerization rates from simple MW vs time datasets. We used M_w data to calculate k_{app} to mitigate variability we observed in measuring M_n values arising from calibration drift or when comparing data from different SEC instruments. Specifically, we found M_n to be highly sensitive to the concentration of monomers and other lower oligomers that form during depolymerization, species that are difficult to quantify by SEC. While we acknowledge the assumption of a Kuhn distribution across the complete depolymerization timescale is likely an oversimplification, we point out that calculation of k_{app} values from M_w data enables more accurate and reproducible *relative* comparisons of reaction rates under different initial experimental conditions.

Supplemental Methods & Procedures

Supplemental Methods:

Size-Exclusion Chromatography. Size exclusion chromatography (SEC) analysis was performed using a Tosoh EcoSEC Elite system equipped with two Tosoh TSKgel Super AWM-H columns (15 cm x 6 mm ID, 9 µm pore size), and integrated differential refractive index (RI) detector, and a Tosoh Lens3 MALS detector operating at three angles (10 °, 90 °, and 170 °). The mobile phase used was THF (HPLC grade) at a flow rate of 0.3 mL min^{-1}. SEC analysis was also performed on an Agilent 1260 infinity II LC system equipped with 3 PolyPore (250 x 4.6 mm) columns in series and a PolyPore (50 x 4.6 mm) guard column. Detection was conducted using the integrated multidetector suite consisting of a differential RI detector, a multi-angle light scattering detector operating at two angles (15 ° and 90 °), and a differential viscometer. The mobile phase used was THF (HPLC grade) at a flow rate of 0.3 mL min^{-1}.

NMR spectroscopy. ^1H and ^{13}C NMR spectroscopy were recorded on a Bruker Ascend Evo 400 spectrometer operating at 400 MHz equipped with a SampleCase autosampler carousel. Chemical shifts are reported in parts per million (ppm) relative to residual protonated solvent for ^1H ($CHCl_3$, $\delta = 7.26$) and relative to carbon resonances of the solvent for ^{13}C ($CDCl_3$, $\delta = 77.16$).

Thermogravimetric Analysis. Thermogravimetric analysis (TGA) of the various polymer samples was acquired on a TA Instruments TGA 55 and were heated up to 600 °C at a rate of 10 °C/min under N_2 atmosphere.

Differential Scanning Calorimetry. Thermal transitions were determined for the various polymer samples (1-15 mg in Tzero pans) using a TA Instruments Discovery Series DSC 250 by heating the sample from 0 °C to 160 °C at a heating and cooling rate of 25 °C/min (for polycarbonate oligomers) or from -90 ºC to 200 ºC at a heating and cooling rate of 10 ºC/min (for polyurethane samples) under N_2 atmosphere over 3 consecutive heating and cooling cycles. The glass transition (T_g) was taken as the inflection point in the DSC trace of the 2nd heating cycle.

Tensile Measurements. Solutions for polymer solvent casting were prepared by dissolving the appropriate polymer in $CHCl_3$ (12-15 wt% of polymer). The polymer solution was then passed through a small plug of glass wool and cast into a Teflon dish, covered with perforated aluminum foil, and left to evaporate at room temperature. The cast sample was then dried further in a vacuum over at 60 °C for at least 6 h. The resulting polymer film was delaminated from the Teflon dish and small portions of the edges were analyzed via TGA to confirm the absence of residual solvent. Dogbone samples for tensile measurements were prepared from these films by punching with a mini dogbone die. The width, length and thickness of these films was measured using a caliper and tensile strength was measured on an Instron machine (Instron 3343 Single Column Testing System). ASTM D1708 standard with a stretching rate of 1.00 mm/s was used for all measurements.

Dynamic Mechanical Analysis. Dynamic mechanical analysis (DMA) was carried out by a TA Instruments DMA 850 using a tension clamp. Polymer film samples were prepared following the same methodology for tensile measurements and cut into a rectangular shape. Samples were tested in a temperature range from -100 °C to 150 °C at a rate of 3.0 °C/min with a frequency of 1.0 Hz and 15.0 μm amplitude.

Automated Analysis of SEC Data. We developed an object-oriented approach to automate the analysis of batches of SEC data using Python. Our core methods are encapsulated within a base `Chromatograph` object, which provides functions for baseline correction, peak identification, and peak integration. `Chromatograph` takes as input raw chromatographic data for both the sample and a 'blank' and an analysis window specifying the region (in time or volume) within the data where the signals of interest are expected and provides detected peak parameters as outputs including location, height, full width at half maximum (FWHM), and area. The core algorithmic steps are shown in Figure S1 and are discussed in detail below.

1. **Estimation and correction for a variable baseline**. Raw chromatography data is first subjected to baseline correction using the rolling standard deviation distribution according to a method developed by Wang et al.[2] and implemented within the `pybaselines` package in Python. A blank trace is then subtracted, if provided, to suppress nascent baseline artifacts systemic to all analyzed samples. Finally, negative peaks are zeroed to improve peak identification and fitting performance. Our processing assumes that analyte signals exhibit positive reactive index responses.

2. **Peak identification**. Analyte peaks are identified in the baseline-processed data via comparison of local maxima to neighboring values. Comparisons are made based on the prominence of each peak as defined by the vertical distance between the peak maximum and its lowest contour line. In brief, a horizontal line is extended from the peak maximum until it reaches the data window or intersects with adjacent signal. The minima between the peak maximum and these intersection points define the peak bases, the highest of which is assigned as the lowest contour line. Prominence-based peak finding is implemented in the `SciPy` API for Python and is used by `Chromatograph`.

3. **Peak fitting and convolution**. For an assigned data window with N peaks, `Chromatograph` fits a convolution of N amplitude-weighted skew normal distributions to the baseline-corrected signal:

$$S(t) = \frac{A}{\sigma\sqrt{2\pi}} exp\left[-\frac{(t-\tau)^2}{2\sigma^2}\right]\left[1 + erf\left(\frac{\alpha(t-\tau)}{\sigma\sqrt{2}}\right)\right] \tag{S1}$$

where A is the amplitude of the peak, t is the specific time (or volume) point, τ is the peak center, σ is the spread of the peak, and α is the peak skewness. Optimization of the peak fitting values is conducted using least squares minimization implemented in the `SciPy` API. `Chromatograph` stores internally the isolated fits for each identified peak, the convolved fit, and the best fit values for each above parameter.

4. **Calculation of distribution averages**. Molecular weight distribution averages for polymer samples are then calculated from the processed `Chromatograph` via the `SEC_Chromatograph` sub-object, which supplies functions to convert raw time (or volume) data into molecular weight values using a third-order polynomial of the form

$$M(t) = a_3t^3 + a_2t^2 + a_1t + a_0 \tag{S2}$$

where a_3, a_2, a_1, and a_0 are instrument-specific coefficients obtained via SEC calibration with narrow polymethylmethacrylate (PMMA) standards. Number average molecular weight, M_n, weight average molecular weight, M_w, and dispersity, $Đ$, are calculated from infinitesimally small slices of the convoluted fit as follows

$$M_n = \frac{\sum_i S(t)_i}{\sum_i \frac{S(t)_i}{M(t)_i}} \tag{S3}$$

$$M_w = \frac{\sum_i S(t)_i M(t)_i}{\sum_i S(t)_i} \tag{S4}$$

$$Đ = \frac{M_w}{M_n} \tag{S5}$$

5. **Histogram output**. `SEC_Chromatograph` converts one or more of the fitted curves into a histogram to visualize the molecular weight distribution and to provide an input for kinetic Monte Carlo (kMC) simulations (*vide infra*). First, the total concentration of chains is calculated as the denominator of Eq. S3. A specified scaling factor is then applied to approximately target the desired number of chains in the population. The results are output

in the form of a `DataFrame` containing the average degree of polymerization for each distribution slice, $X_{n,i}$, and the number of chains of $X_{n,i}$ as columns.

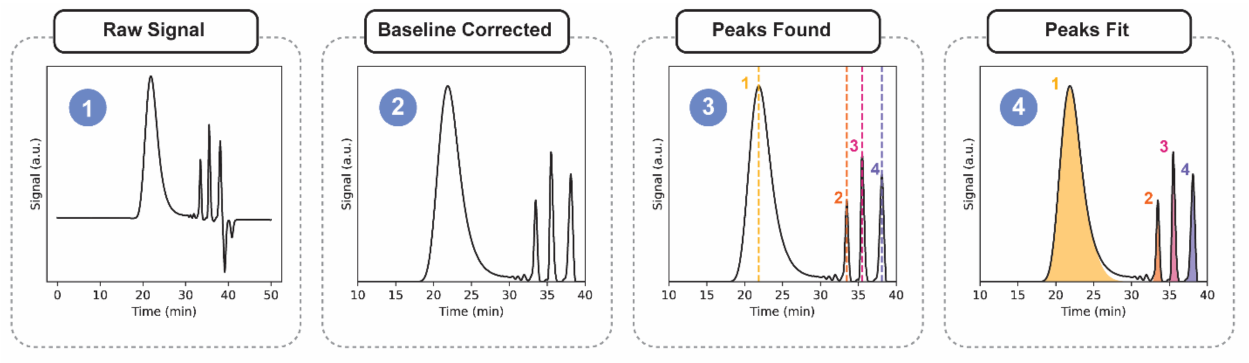


Figure S1. Baseline correction, peak identification, and peak fitting algorithms implemented in Chromatograph as applied to real SEC data. Shown from left to right are: (1) the raw data trace obtained from the SEC instrument; (2) the data after baseline correction; (3) the peaks identified in the signal; and (4) the individual fits of the identified peaks.

INTERSECT Cloud-Computing. A service for intermediate storage of data was hosted within the ORNL Research Cloud.

Kinetic Monte Carlo Simulations. To extract mechanistic information from depolymerization kinetics, the depolymerizations are modeled as an ensemble of elementary reactions. We neglect solvent effects (i.e., we assume that all components are fully soluble), changes in solution viscosity (i.e., we assume dilute monomer and polymer concentrations, approximately at or below 1 M), and changes in temperature (i.e., we assume that temperature is constant throughout the reaction duration). Rate equations are expressed in pseudo-first order fashion with respect to polymer chains, assuming that changes in nucleophile and catalyst concentration are negligible. Simulations are performed in a unit system where rate constants for bond scission are on the order of 1.[3] This allows for the system of reactions for polymer chain scission to be described by a single input parameter $r = \frac{k_{rand}}{k_{end}}$ that describes the relative rate constants of random and chain-end scission (k_{rand} and k_{end}, respectively) and drastically simplifies, and increases the accuracy of, the fitting of experimental data. The molecular weight dependencies of these rate constants are not considered; however, as we note in the main text, this is likely an oversimplification to be addressed in future simulation implementations.

Simulation sizes are restricted to an upper bound of N_{chains} (typically $10^2 – 10^3$) to limit calculation costs. For each simulation, an initial population of chains is 'synthesized' from real experimental data by downsampling of a X_n frequency histogram generated from our automated data analysis routine (*vide supra*). An example is provided in Figure S2. The overall depolymerization conversion, p, is calculated by tracking the change in the total number of bonds

$$p = 1 - \frac{B}{B_0} \tag{S6}$$

where B and B_0 represent the instantaneous and initial number of bonds, respectively. At various intervals of p, molecular weight distribution averages are calculated for the population of chains and stored for later analysis. Simulations are terminated when the specified target p is achieved.

1. **Choosing a reaction**. The kMC simulation is based on an algorithm developed by Gillespie where discreet reaction events are stochastically sampled.[4] The probably of a reaction j occurring at a given instant is given by

   $$p_j = \frac{R_j}{\sum_j R_j} \tag{S6}$$

   where R_j is the rate of reaction j. At each simulation step, a reaction is selected by sampling all reaction probabilities. A random number, α, uniformly distributed between 0-1 is generated and compared against the cumulative reaction probabilities such that

   $$\sum_{j}^{\mu-1} p_j < \alpha < \sum_{j}^{\mu} p_j \tag{S7}$$

   where μ is the index of the selected reaction. The selected reaction is executed as described in the sections below, after which reaction probabilities are recalculated based on the outcome of the reaction and used for the subsequent simulation step.

2. **Random chain scission**. For this type of reaction, the overall number (concentration) of bonds is used as an input. A bond is selected at random from the population, and the outcome of breaking this bond is evaluated, generating two new chains of length $L_1 = L_0 -$

x and $L_2 = x - 1$, where L_1 and L_2 represent the lengths of the new chains in terms of the number of bonds, L_0 is the number of bonds initially present in the selected chain, and x is a random bond location. The probability of a random scission event is calculated from the stipulated rate expression

$$Rate = k_{rand}[B] \tag{S8}$$

where k_{rand} is the rate constant for random chain scission and $[B]$ is the instantaneous concentration of bonds.

3. **Chain end scission**. For this type of reaction, the overall number (concentration) of polymer chain ends is used as an input. A bond located at a chain end is selected randomly from the population, and the outcome of breaking this bond is evaluated, generating one monomer ($L_1 = 0$) and one chain of length $L_2 = L_0 - 1$. The probability of a chain end scission event is calculated from the stipulated rate expression

$$Rate = k_{end}[B_e] \tag{S9}$$

where k_{end} is the rate constant for chain end scission and $[B_e]$ is the instantaneous concentration of bonds located at polymer chain ends.

4. **Calculation of molecular weight distribution averages**. Molecular weight distribution averages are calculated from the population of chains at various conversion intervals using the standard equations (and Eq. S5 above)

$$M_n = \frac{\sum M_i N_i}{\sum N_i} \tag{S10}$$

$$M_w = \frac{\sum M_i^2 N_i}{\sum M_i N_i} \tag{S11}$$

where N_i is the number of chains of mass M_i, calculated as $M_i = M_0(B_i + 1)$ with M_0 being the mass of the polymer repeat unit (specified as an input to the simulation) and B_i the number of bonds in chain i.

5. **Simulation implementation**. Algorithms for kMC simulations are implemented within a `PolymerSample` object in Python. The object is initiated by passing the polymer's repeat unit molecular weight as a parameter. The population of polymer chains used for the simulation is generated from a X_n frequency histogram using `build_from_list()` where the frequency array and the desired simulation size, in terms of the number of chains, are passed as arguments (Figure S3). Calling `PolymerSample.depolymerize()` with a stipulated target p and input reaction rate constant ratio r executes the simulation and returns an `np.array` containing information about the conversion-dependent changes in molecular weight distribution averages. To fit experimental data, multiple simulations are run, varying the input r, until a minimum mean squared error value was obtained.

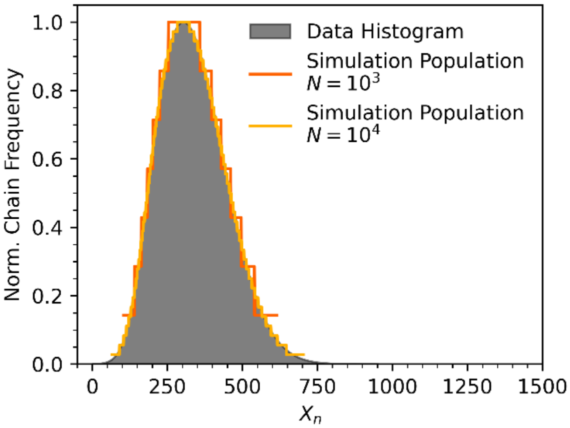


Figure S2. Comparison of chain population histogram, generated from experimental data and passed to `PolymerSample` as an argument, against the initial population of chains generated by the simulation as a function of N.

6. **Simulation benchmarking, accuracy, and precision**. Simulations are run with different N to determine the optimal simulation size in terms of computational cost, accuracy, and precision. Five replicates are run for each N, targeting low overall $p = 0.05$ where the variability in simulation outcomes is found to be highest. As shown in Figure S3, $N = 10^3$ is sufficient to minimize the number of reaction cycles needed to complete the simulation, maximize the simulation accuracy as determined by comparison to the expected value (from theory, *vide infra*), and maximize precision as judged by the reduction in variance between replicates.

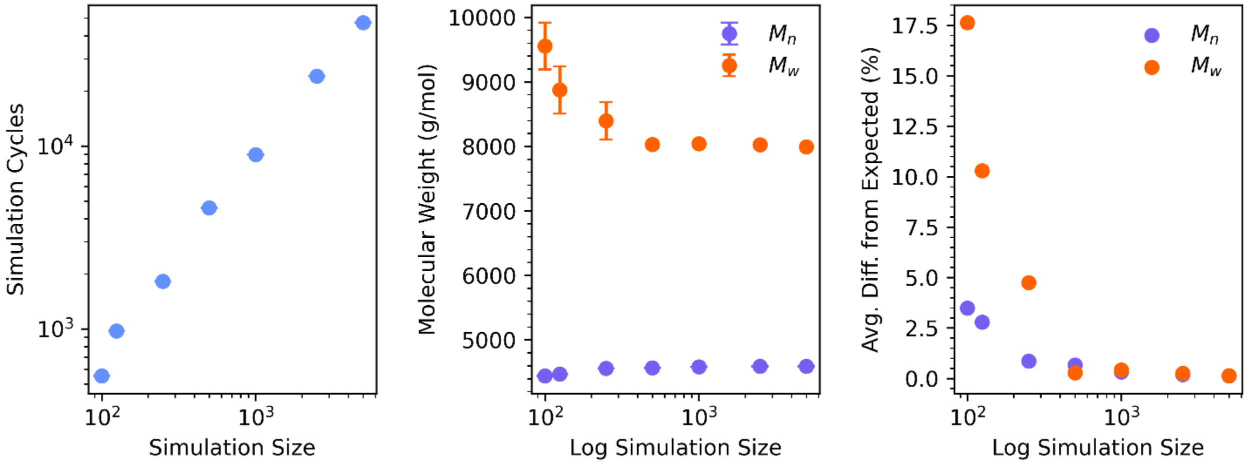


Figure S3. (Left) Average number of simulation cycles required to reach target $p = 0.05$ as a function of simulation size, N. (Middle) Simulated M_n and M_w as a function of N. The error bars represent the standard deviation of the average values across $n = 5$ replicates. (Right) Average percentage difference in simulated molecular weight and expected molecular weight as a function of N. The expected molecular weight values were calculated using Eqs. S16 and S17.

Model Calibration and Evaluation. Observed and simulated datasets were analyzed by estimating model parameters through a nonlinear least-squares (NLLS) optimization using the `leastsq` function from the `scipy.optimize` module. In cases where the model was fixed and required no parameter estimation, goodness-of-fit was evaluated by calculating the coefficient of determination **(**R^2**)** and the mean squared error (MSE) directly from the residuals.

Statistical Assessment of Precision. The intra-experiment precision for both automated and manual kinetic experiments was evaluated by calculating the percent coefficient of variation (%CV) from $n = 3$ technical replicates per time point. A Welch's t-test was used to compare the **mean** %CV of manual and automated depolymerization reactions ($n = 21$ per group) to evaluate differences in measurement precision ($\alpha = 0.05$). To satisfy the assumption of normality, %CV values were log-transformed prior to analysis.

Statistical Comparison of Regression Slopes. The homogeneity of regression slopes across datasets was evaluated by testing the interaction effect between the categorical dataset indicator and the continuous covariate (time) via analysis of covariance (ANCOVA). A statistically significant interaction indicated heterogeneous slopes, signifying that the relationship between time and the dependent variable differed between groups.

Supplemental Procedures:

Table S1. Summary of kinetic experiments included in this work.

Run	Automated?	[EG] (M)	[mTBD] (M)
1	Yes	0.29	0.0049
2	Yes	0.29	0.0049
3	Yes	0.29	0.0049
4	No	0.29	0.0049
5	No	0.29	0.0049
6	No	0.29	0.0049
7	Yes	0.29	0.00098
8	Yes	0.29	0.0098
9	Yes	0.193	0.0049
10	Yes	0.384	0.0049

**PC concentration was fixed at 0.097 M with respect to repeat units for all experiments and the temperature was held at 25 °C.*

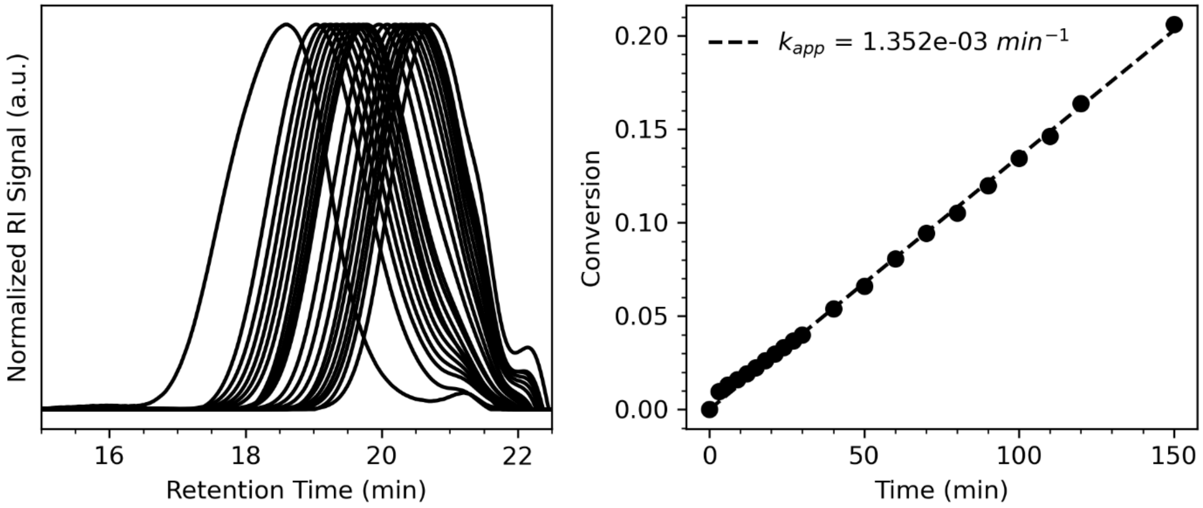
Representative Manual PC Depolymerization. To a 100 mL round-bottom flask equipped with a stir bar, 84 mL of THF were added. The flask was submerged in an oil bath heated to 40 °C and then 2.1 g (8.26 mmol) of PC pieces were added slowly over the course of approximately 5 hours. During this time, twenty two-dram vials were charged with 0.5 mL of a solution of AcOH (8.58 mmol) in THF (0.4 M). Upon complete dissolution, the PC stock solution was allowed to cool to room temperature. EG (1.39 mL; 24.8 mmol) was then added dropwise with vigorous stirring. An initial t_0 aliquot (4.0 mL) was collected from the stock solution and added to a 2-dram vial in the absence of AcOH. Catalytic mTBD (56.5 µL; 0.393 mmol) was then rapidly injected into the PC:EG solution and a stopwatch was started. A 3.5 mL aliquot of the reaction mixture was collected and immediately quenched by addition to one of the 2-dram vials charged with AcOH at each of the following time points (in min): 3, 6, 9, 12, 15, 18, 21, 24, 27, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 150.

Polyurethane Synthesis and Film Casting. To separate oven-dried 20 mL scintillation vials equipped with stir bars, poly(carbonate) BPA (PC) (0.254 mmol) and hydroxy-terminated hydrogenated polybutadiene (PBD) (0.254 mmol) were added. The vials were transferred into an

N_2 filled glovebox and anhydrous THF was added to each (4 mL to dissolve PC, and 12 mL to dissolve PBD). In a separate oven-dried 20 mL scintillation vial, dibutyltin dilaurate (DBTDL; 3.0 uL, 0.00508 mmol) and hexamethylenediisiocyanate (HDI; 83 uL, 0.52 mmol) were dissolved in anhydrous THF (2 mL). The DBTDL/HDI solution was transferred to an oven-dried 50 mL Schlenk flask equipped with a stir bar and the PC solution was added dropwise with stirring. The reaction flask was sealed with a rubber septum and transferred to a Schlenk line, then placed in an oil bath and equipped with a Findenser under positive N_2 flow. The reaction mixture was then heated to reflux (70 ºC) and allowed to stir for two hours under N_2. After two hours, the PBD solution was added dropwise and the reaction mixture was allowed to stir at 65 ºC for 16 h under N_2.

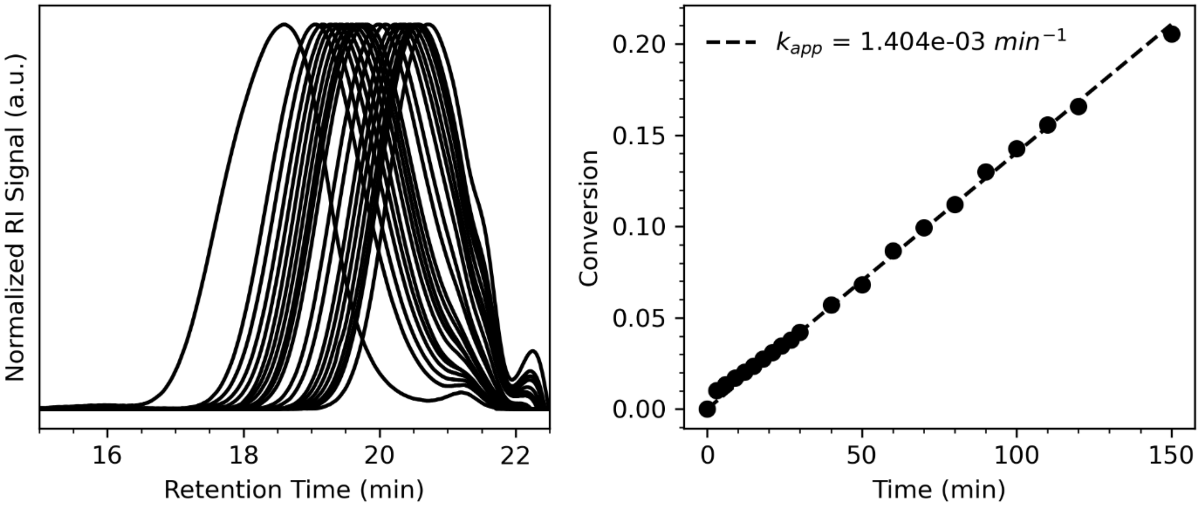
The resulting polymer was isolated by 3x precipitations into chilled methanol. After the final precipitation, the polymer was redissolved in 30 mL of DCM and transferred to a 40 mL scintillation vial. The DCM was evaporated and then the polymer residue was dried under vacuum.

Supplemental Experimental Data



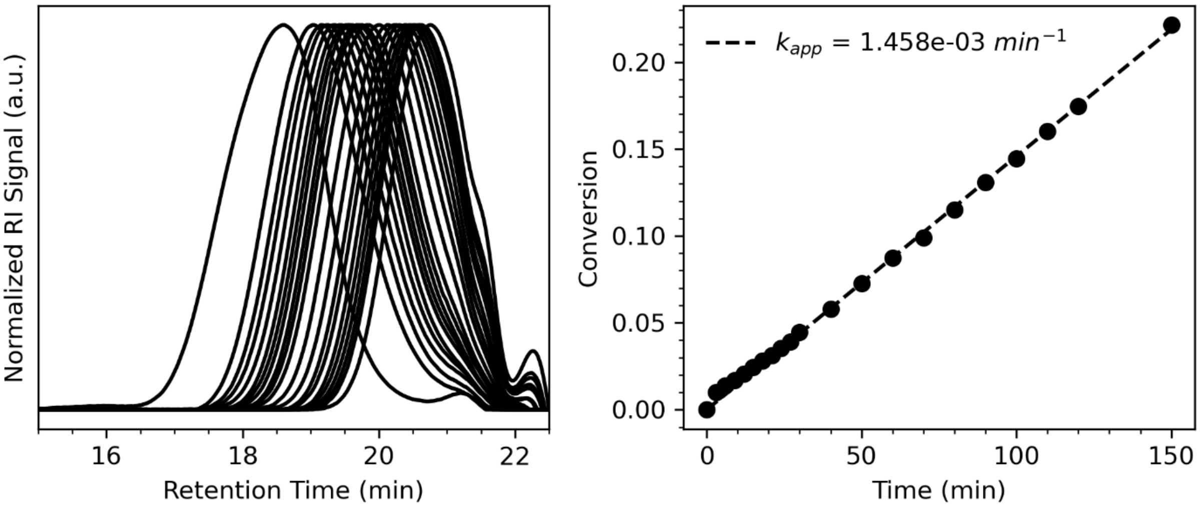
Time	Pks. Found	Pk. Center	Pk. Height	X_n	X_w	M_n	M_w	Đ
0.0	1.0	18.593	18.593	124.05	242.92	31546.0	61774.0	1.96
3.0	1.0	19.033	19.033	43.91	110.6	11167.0	28126.0	2.52
6.0	1.0	19.165	19.165	30.96	92.07	7873.0	23413.0	2.97
9.0	1.0	19.248	19.248	25.19	81.03	6406.0	20606.0	3.22
12.0	1.0	19.333	19.333	20.5	71.73	5214.0	18241.0	3.5
15.0	1.0	19.422	19.422	16.98	64.01	4317.0	16278.0	3.77
18.0	1.0	19.515	19.515	14.02	57.13	3566.0	14527.0	4.07
21.0	1.0	19.598	19.598	13.63	51.62	3465.0	13126.0	3.79
24.0	1.0	19.657	19.657	11.4	47.42	2899.0	12059.0	4.16
27.0	1.0	19.723	19.723	9.18	43.72	2334.0	11117.0	4.76
30.0	1.0	19.775	19.775	8.52	40.83	2166.0	10384.0	4.79
40.0	1.0	19.96	19.96	6.15	31.68	1563.0	8056.0	5.15
50.0	1.0	20.082	20.082	5.22	26.39	1327.0	6712.0	5.06
60.0	1.0	20.197	20.197	4.29	22.05	1090.0	5607.0	5.14
70.0	1.0	20.287	20.287	4.21	19.15	1070.0	4870.0	4.55
80.0	1.0	20.358	20.358	3.54	17.3	901.0	4399.0	4.88
90.0	1.0	20.43	20.43	3.18	15.31	809.0	3894.0	4.81
100.0	1.0	20.5	20.5	2.76	13.73	701.0	3492.0	4.98
110.0	1.0	20.55	20.55	2.59	12.69	658.0	3228.0	4.91
120.0	1.0	20.613	20.613	2.43	11.4	619.0	2899.0	4.68
150.0	2.0	20.733	20.733	1.72	9.15	437.0	2327.0	5.32

Figure S4. Output from Python batch analysis of kinetic samples with initial conditions: [mTBD] = 0.0049 M; [EG] = 0.29 M (**Run 1**). Peak center and peak height correspond to the leftmost identified peak.



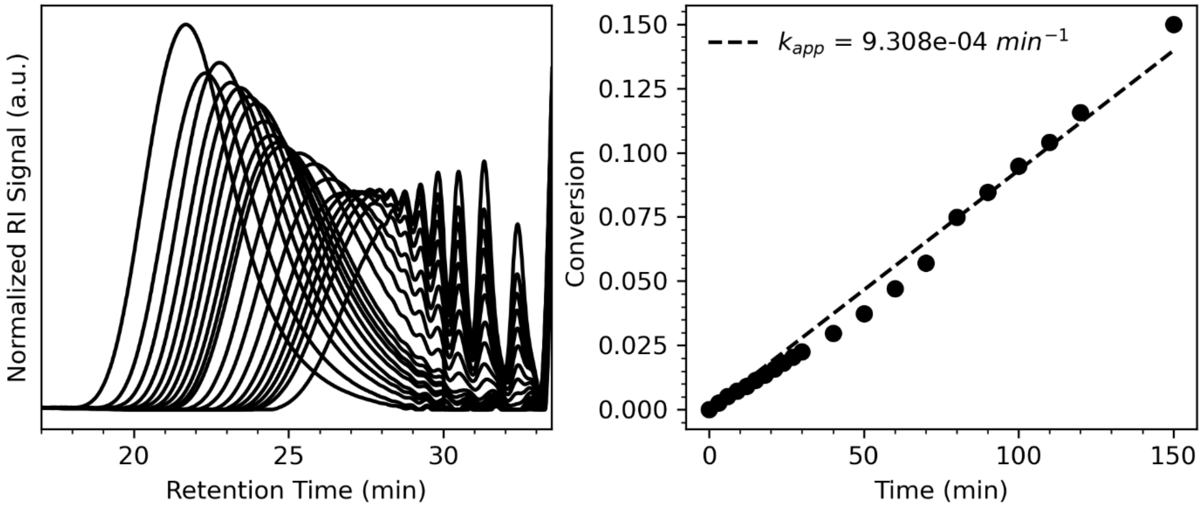
Time	Pks. Found	Pk. Center	Pk. Height	X_n	X_w	M_n	M_w	Đ
0.0	1.0	18.593	18.593	124.05	242.92	31546.0	61774.0	1.96
3.0	1.0	19.067	19.067	43.04	108.11	10946.0	27493.0	2.51
6.0	1.0	19.16	19.16	33.5	91.54	8519.0	23279.0	2.73
9.0	1.0	19.272	19.272	25.68	78.08	6531.0	19857.0	3.04
12.0	1.0	19.363	19.363	21.37	68.87	5434.0	17513.0	3.22
15.0	1.0	19.422	19.422	19.54	62.2	4970.0	15817.0	3.18
18.0	1.0	19.523	19.523	15.21	55.17	3868.0	14029.0	3.63
21.0	1.0	19.592	19.592	12.84	50.02	3264.0	12720.0	3.9
24.0	1.0	19.665	19.665	12.06	45.92	3068.0	11678.0	3.81
27.0	1.0	19.728	19.728	10.06	42.62	2559.0	10837.0	4.23
30.0	1.0	19.815	19.815	8.67	38.93	2206.0	9900.0	4.49
40.0	1.0	19.988	19.988	6.27	30.03	1595.0	7637.0	4.79
50.0	1.0	20.085	20.085	5.66	25.69	1439.0	6533.0	4.54
60.0	1.0	20.22	20.22	4.69	20.66	1193.0	5254.0	4.4
70.0	1.0	20.32	20.32	3.85	18.22	980.0	4633.0	4.73
80.0	1.0	20.372	20.372	3.58	16.31	910.0	4148.0	4.56
90.0	2.0	20.452	20.452	3.92	14.18	998.0	3606.0	3.61
100.0	2.0	20.513	20.513	2.74	12.98	697.0	3300.0	4.73
110.0	2.0	20.572	20.572	2.53	11.95	644.0	3040.0	4.72
120.0	2.0	20.572	20.572	2.05	11.26	522.0	2863.0	5.48
150.0	2.0	20.728	20.728	1.77	9.17	449.0	2333.0	5.2

Figure S5. Output from Python batch analysis of kinetic samples with initial conditions: [mTBD] = 0.0049 M; [EG] = 0.29 M (**Run 2**). Peak center and peak height correspond to the leftmost identified peak.



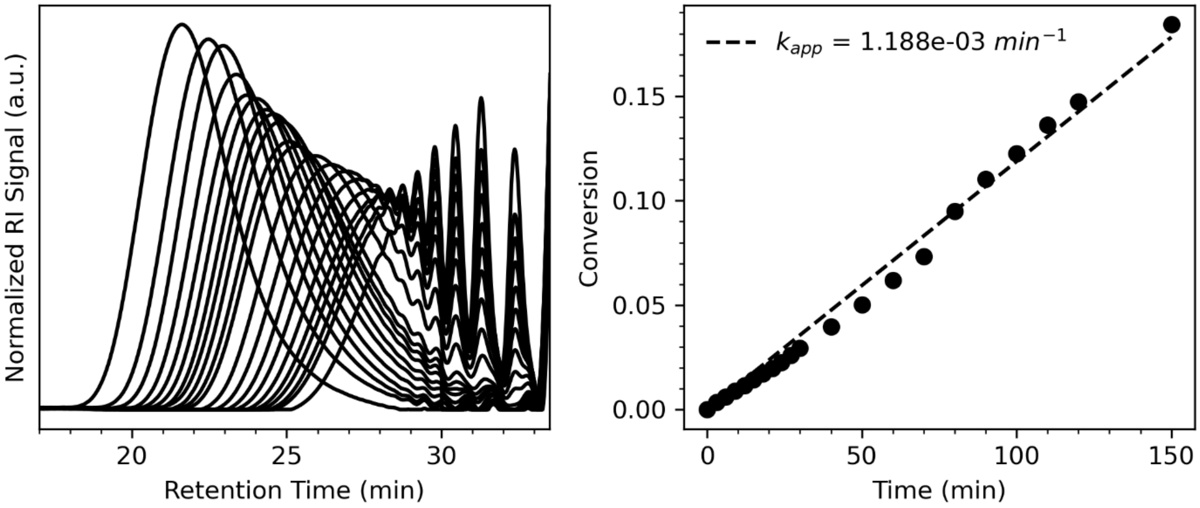
Time	Pks. Found	Pk. Center	Pk. Height	X_n	X_w	M_n	M_w	Đ
0.0	1.0	18.593	18.593	124.05	242.92	31546.0	61774.0	1.96
3.0	1.0	19.04	19.04	41.71	108.7	10608.0	27643.0	2.61
6.0	1.0	19.157	19.157	30.44	89.81	7741.0	22838.0	2.95
9.0	1.0	19.242	19.242	23.06	78.8	5863.0	20040.0	3.42
12.0	1.0	19.36	19.36	18.62	68.54	4734.0	17430.0	3.68
15.0	1.0	19.422	19.422	16.54	60.35	4207.0	15348.0	3.65
18.0	1.0	19.527	19.527	13.29	54.18	3379.0	13778.0	4.08
21.0	1.0	19.595	19.595	11.62	49.77	2956.0	12657.0	4.28
24.0	1.0	19.68	19.68	10.81	45.17	2750.0	11487.0	4.18
27.0	1.0	19.732	19.732	9.05	41.49	2301.0	10551.0	4.59
30.0	1.0	19.85	19.85	8.0	37.24	2035.0	9469.0	4.65
40.0	1.0	19.995	19.995	6.19	29.74	1574.0	7564.0	4.81
50.0	1.0	20.142	20.142	5.16	24.22	1312.0	6160.0	4.7
60.0	1.0	20.227	20.227	4.7	20.51	1196.0	5216.0	4.36
70.0	1.0	20.292	20.292	4.15	18.27	1055.0	4647.0	4.4
80.0	1.0	20.393	20.393	3.86	15.89	981.0	4042.0	4.12
90.0	1.0	20.443	20.443	3.19	14.09	810.0	3584.0	4.42
100.0	1.0	20.525	20.525	2.97	12.85	756.0	3267.0	4.32
110.0	1.0	20.587	20.587	2.7	11.64	686.0	2961.0	4.32
120.0	1.0	20.615	20.615	2.52	10.72	641.0	2726.0	4.25
150.0	2.0	20.747	20.747	1.68	8.54	427.0	2171.0	5.08

Figure S6. Output from Python batch analysis of kinetic samples with initial conditions: [mTBD] = 0.0049 M; [EG] = 0.29 M (**Run 3**). Peak center and peak height correspond to the leftmost identified peak.



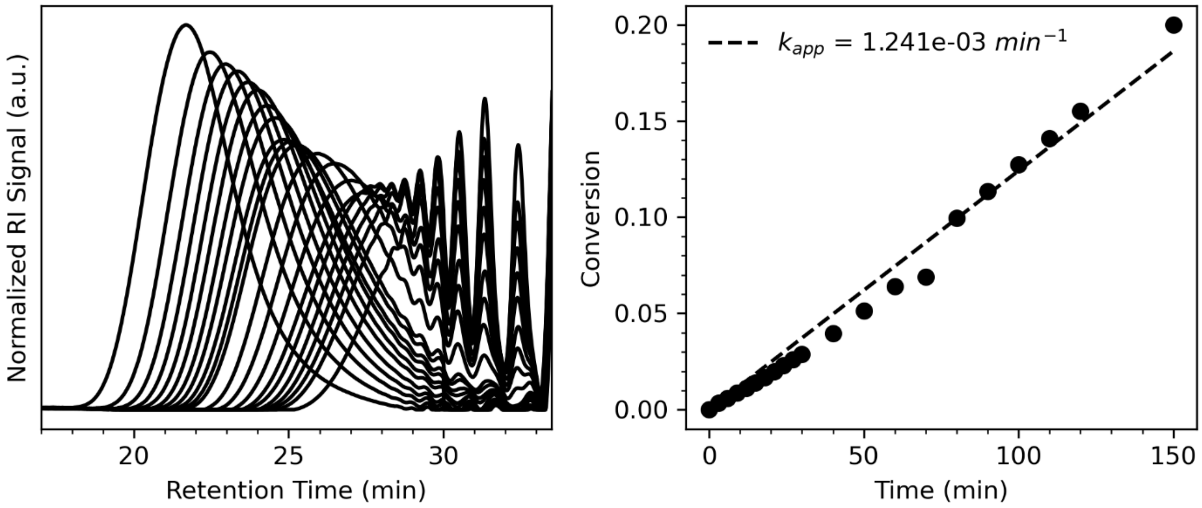
Time	Pks. Found	Pk. Center	Pk. Height	X_n	X_w	M_n	M_w	Đ
0.0	2.0	21.683	21.683	165.76	278.15	42153.0	70734.0	1.68
3.0	2.0	22.283	22.283	113.67	192.42	28907.0	48932.0	1.69
6.0	2.0	22.767	22.767	88.26	150.91	22445.0	38376.0	1.71
9.0	2.0	23.117	23.117	72.99	126.78	18561.0	32241.0	1.74
12.0	2.0	23.417	23.417	62.9	110.0	15995.0	27974.0	1.75
15.0	2.0	23.7	23.7	53.6	96.13	13630.0	24445.0	1.79
18.0	2.0	23.933	23.933	47.69	85.99	12127.0	21867.0	1.8
21.0	2.0	24.2	24.2	41.24	76.61	10487.0	19481.0	1.86
24.0	2.0	24.433	24.433	37.14	69.28	9445.0	17618.0	1.87
27.0	2.0	24.633	24.633	33.26	63.42	8457.0	16128.0	1.91
30.0	2.0	24.817	24.817	30.09	58.93	7651.0	14987.0	1.96
40.0	2.0	25.367	25.367	22.61	46.93	5750.0	11934.0	2.08
50.0	2.0	25.767	25.767	17.25	38.66	4387.0	9830.0	2.24
60.0	2.0	26.3	26.3	13.22	31.6	3363.0	8035.0	2.39
70.0	2.0	26.8	26.8	10.79	26.58	2744.0	6760.0	2.46
80.0	7.0	27.1	27.1	5.49	20.76	1396.0	5279.0	3.78
90.0	7.0	27.35	27.35	5.25	18.53	1334.0	4712.0	3.53
100.0	8.0	27.633	27.633	5.0	16.64	1272.0	4231.0	3.33
110.0	8.0	27.95	27.95	4.85	15.24	1233.0	3875.0	3.14
120.0	8.0	28.333	28.333	4.56	13.8	1159.0	3509.0	3.03
150.0	8.0	28.367	28.367	3.96	10.77	1008.0	2738.0	2.72

Figure S7. Output from Python batch analysis of manually collected kinetic samples with initial conditions: [mTBD] = 0.0049 M; [EG] = 0.29 M (**Run 4**). Peak center and peak height correspond to the leftmost identified peak.



Time	Pks. Found	Pk. Center	Pk. Height	X_n	X_w	M_n	M_w	Đ
0.0	2.0	21.617	21.617	169.32	284.12	43057.0	72252.0	1.68
3.0	2.0	22.483	22.483	103.44	176.6	26306.0	44910.0	1.71
6.0	2.0	22.933	22.933	81.41	139.19	20703.0	35395.0	1.71
9.0	2.0	23.383	23.383	65.6	113.71	16681.0	28917.0	1.73
12.0	2.0	23.7	23.7	54.46	96.43	13848.0	24522.0	1.77
15.0	2.0	24.033	24.033	45.6	82.9	11595.0	21081.0	1.82
18.0	2.0	24.333	24.333	39.3	72.97	9994.0	18556.0	1.86
21.0	2.0	24.583	24.583	33.92	65.3	8627.0	16605.0	1.92
24.0	2.0	24.783	24.783	30.11	58.75	7658.0	14941.0	1.95
27.0	2.0	25.117	25.117	26.22	52.47	6669.0	13342.0	2.0
30.0	2.0	25.317	25.317	22.76	47.33	5788.0	12035.0	2.08
40.0	2.0	25.883	25.883	15.8	36.72	4018.0	9338.0	2.32
50.0	2.0	26.4	26.4	12.25	29.82	3115.0	7584.0	2.43
60.0	2.0	26.983	26.983	9.83	24.78	2501.0	6301.0	2.52
70.0	2.0	27.333	27.333	8.31	21.2	2113.0	5390.0	2.55
80.0	8.0	27.933	27.933	5.02	16.61	1277.0	4224.0	3.31
90.0	8.0	28.333	28.333	4.65	14.45	1183.0	3674.0	3.11
100.0	8.0	28.333	28.333	4.46	13.07	1134.0	3323.0	2.93
110.0	8.0	28.333	28.333	4.23	11.79	1075.0	2998.0	2.79
120.0	8.0	28.333	28.333	4.03	10.94	1026.0	2781.0	2.71
150.0	7.0	28.75	28.75	3.59	8.82	912.0	2243.0	2.46

Figure S8. Output from Python batch analysis of manually collected kinetic samples with initial conditions: [mTBD] = 0.0049 M; [EG] = 0.29 M (**Run 5**). Peak center and peak height correspond to the leftmost identified peak.



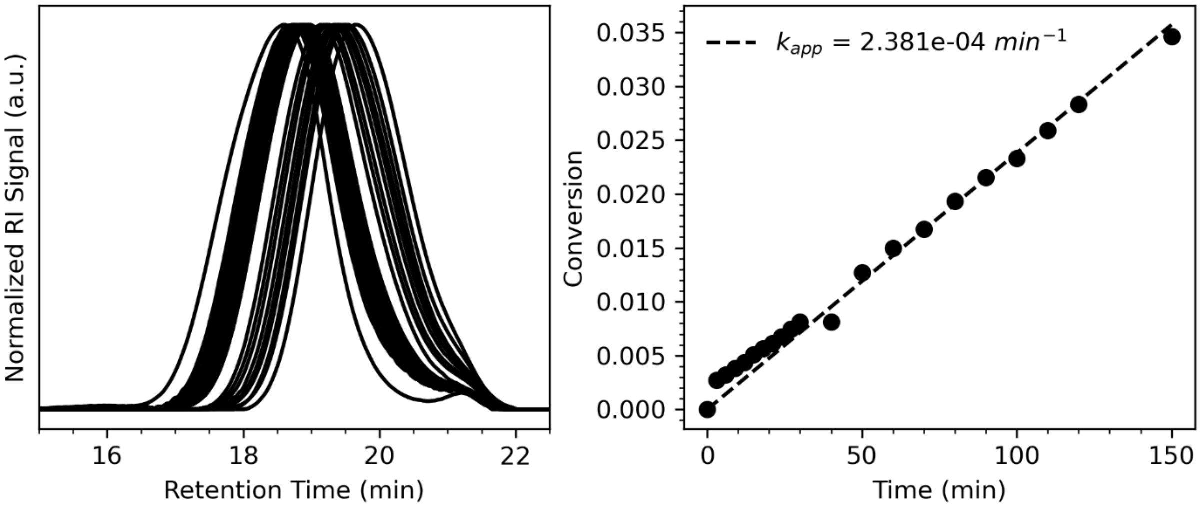
Time	Pks. Found	Pk. Center	Pk. Height	X_n	X_w	M_n	M_w	Đ
0.0	2.0	21.667	21.667	166.2	278.67	42265.0	70865.0	1.68
3.0	2.0	22.433	22.433	105.49	178.75	26826.0	45455.0	1.69
6.0	2.0	22.967	22.967	80.48	138.2	20467.0	35144.0	1.72
9.0	2.0	23.333	23.333	65.67	114.45	16700.0	29105.0	1.74
12.0	2.0	23.7	23.7	54.93	97.33	13969.0	24751.0	1.77
15.0	2.0	24.0	24.0	45.97	83.89	11689.0	21334.0	1.83
18.0	2.0	24.3	24.3	39.75	73.46	10108.0	18682.0	1.85
21.0	2.0	24.583	24.583	34.16	65.1	8686.0	16554.0	1.91
24.0	2.0	24.85	24.85	29.64	58.14	7538.0	14785.0	1.96
27.0	2.0	25.067	25.067	25.7	52.17	6535.0	13268.0	2.03
30.0	2.0	25.233	25.233	23.46	48.32	5967.0	12287.0	2.06
40.0	2.0	25.933	25.933	15.83	36.71	4026.0	9335.0	2.32
50.0	2.0	26.467	26.467	11.8	29.17	3000.0	7418.0	2.47
60.0	2.0	27.033	27.033	9.53	23.96	2423.0	6094.0	2.52
70.0	3.0	27.633	27.633	10.66	22.34	2711.0	5681.0	2.1
80.0	8.0	27.95	27.95	4.92	15.87	1250.0	4036.0	3.23
90.0	8.0	28.333	28.333	4.64	14.02	1181.0	3566.0	3.02
100.0	8.0	28.317	28.317	4.4	12.58	1119.0	3199.0	2.86
110.0	8.0	28.35	28.35	4.11	11.41	1045.0	2902.0	2.78
120.0	7.0	28.767	28.767	3.92	10.4	996.0	2644.0	2.65
150.0	7.0	28.8	28.8	3.37	8.15	857.0	2072.0	2.42

Figure S9. Output from Python batch analysis of manually collected kinetic samples with initial conditions: [mTBD] = 0.0049 M; [EG] = 0.29 M (**Run 6**). Peak center and peak height correspond to the leftmost identified peak.

Table S2. ANCOVA comparing the slopes of linearized plots for automated (runs 1-3) or manual (runs 4-6) kinetic experiments. The analysis shows a significant difference in slopes.

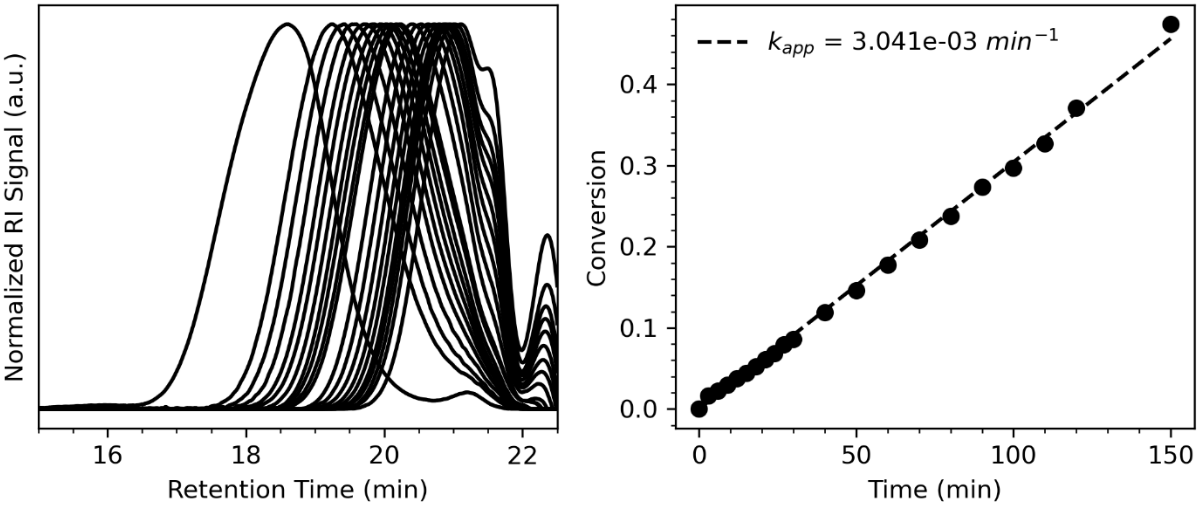
```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      Y   R-squared:                       0.996
Model:                            OLS   Adj. R-squared:                  0.996
Method:                 Least Squares   F-statistic:                     3362.
Date:                Fri, 20 Mar 2026   Prob (F-statistic):           4.05e-44
Time:                        08:04:15   Log-Likelihood:                 171.60
No. Observations:                  40   AIC:                            -335.2
Df Residuals:                      36   BIC:                            -328.4
Df Model:                           3
Covariance Type:            nonrobust
================================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------
Intercept      0.0027      0.001      2.173      0.036       0.000       0.005
Group[T.B]    -0.0093      0.002     -5.316      0.000      -0.013      -0.006
X              0.0014   1.84e-05     74.436      0.000       0.001       0.001
Group[T.B]:X  -0.0002   2.61e-05     -6.352      0.000      -0.000      -0.000
==============================================================================
Omnibus:                       11.719   Durbin-Watson:                   0.468
Prob(Omnibus):                  0.003   Jarque-Bera (JB):               12.292
Skew:                          -1.006   Prob(JB):                      0.00214
Kurtosis:                       4.824   Cond. No.                         276.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```



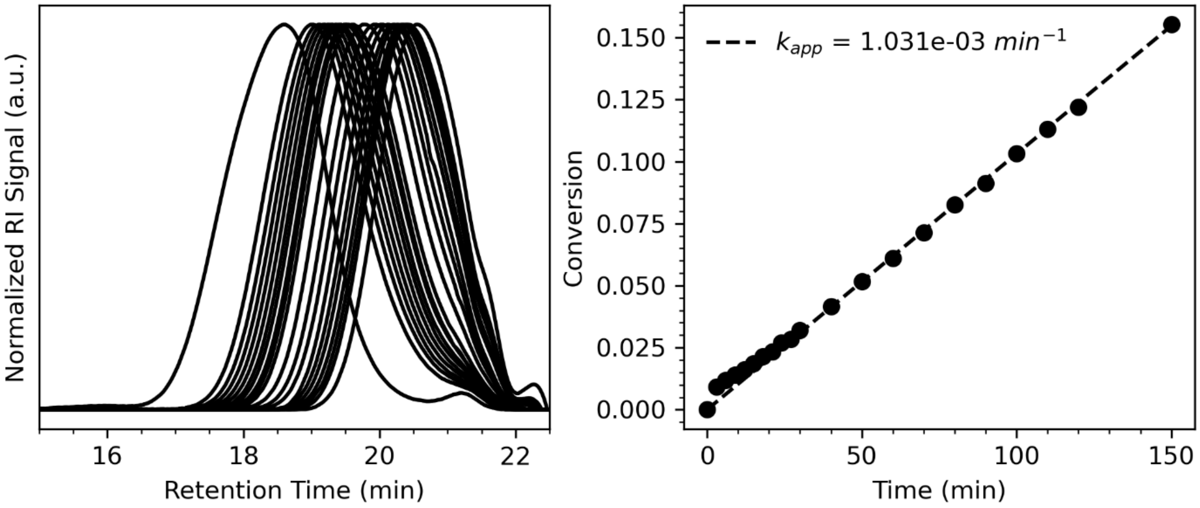
Time	Pks. Found	Pk. Center	Pk. Height	X_n	X_w	M_n	M_w	Đ
0.0	1.0	18.593	18.593	124.05	242.92	31546.0	61774.0	1.96
3.0	1.0	18.737	18.737	81.49	181.71	20724.0	46208.0	2.23
6.0	1.0	18.74	18.74	75.6	173.78	19225.0	44192.0	2.3
9.0	1.0	18.77	18.77	70.38	164.87	17897.0	41926.0	2.34
12.0	1.0	18.802	18.802	65.55	157.06	16669.0	39941.0	2.4
15.0	1.0	18.837	18.837	61.22	148.65	15567.0	37801.0	2.43
18.0	1.0	18.853	18.853	58.51	142.62	14880.0	36269.0	2.44
21.0	1.0	18.888	18.888	54.48	137.88	13854.0	35064.0	2.53
24.0	1.0	18.898	18.898	51.85	132.17	13186.0	33610.0	2.55
27.0	1.0	18.93	18.93	47.62	126.09	12111.0	32065.0	2.65
30.0	1.0	18.982	18.982	45.31	121.07	11522.0	30787.0	2.67
40.0	1.0	18.982	18.982	45.31	121.07	11522.0	30787.0	2.67
50.0	1.0	19.133	19.133	32.3	94.14	8213.0	23941.0	2.92
60.0	1.0	19.212	19.212	26.21	85.0	6665.0	21616.0	3.24
70.0	1.0	19.253	19.253	22.95	79.03	5837.0	20098.0	3.44
80.0	1.0	19.338	19.338	19.34	71.4	4917.0	18158.0	3.69
90.0	1.0	19.38	19.38	17.63	66.13	4483.0	16816.0	3.75
100.0	1.0	19.407	19.407	15.63	62.44	3975.0	15878.0	3.99
110.0	1.0	19.457	19.457	15.68	57.62	3988.0	14653.0	3.67
120.0	1.0	19.537	19.537	13.07	53.81	3323.0	13684.0	4.12
150.0	1.0	19.645	19.645	10.93	45.88	2779.0	11668.0	4.2

Figure S10. Output from Python batch analysis of kinetic samples with initial conditions: [mTBD] = 0.00098 M; [EG] = 0.29 M (**Run 7**). Peak center and peak height correspond to the leftmost identified peak.



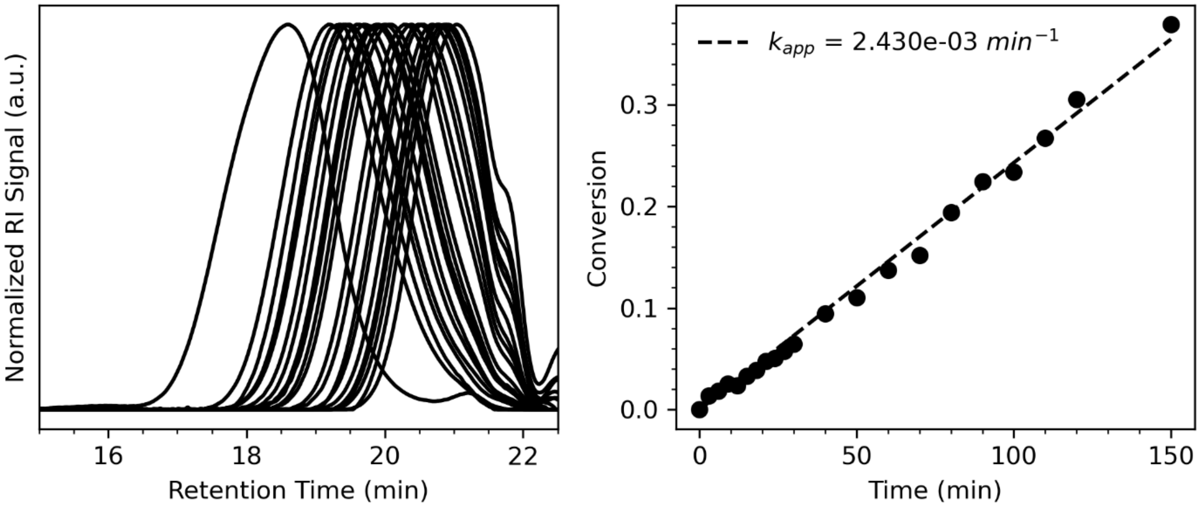
Time	Pks. Found	Pk. Center	Pk. Height	X_n	X_w	M_n	M_w	Đ
0.0	1.0	18.593	18.593	124.05	242.92	31546.0	61774.0	1.96
3.0	1.0	19.237	19.237	23.7	80.29	6027.0	20417.0	3.39
6.0	1.0	19.418	19.418	17.31	64.39	4403.0	16374.0	3.72
9.0	1.0	19.597	19.597	12.8	52.22	3255.0	13280.0	4.08
12.0	1.0	19.73	19.73	9.82	42.8	2498.0	10884.0	4.36
15.0	1.0	19.847	19.847	8.05	37.35	2046.0	9499.0	4.64
18.0	1.0	19.945	19.945	7.22	32.31	1837.0	8217.0	4.47
21.0	1.0	20.028	20.028	6.48	28.33	1649.0	7205.0	4.37
24.0	1.0	20.093	20.093	5.56	25.57	1415.0	6503.0	4.6
27.0	1.0	20.173	20.173	4.73	22.46	1202.0	5711.0	4.75
30.0	1.0	20.238	20.238	4.52	20.85	1149.0	5302.0	4.61
40.0	1.0	20.39	20.39	3.5	15.42	889.0	3922.0	4.41
50.0	1.0	20.533	20.533	2.85	12.68	725.0	3225.0	4.45
60.0	1.0	20.64	20.64	2.28	10.54	581.0	2680.0	4.61
70.0	2.0	20.718	20.718	1.71	9.06	434.0	2303.0	5.31
80.0	2.0	20.803	20.803	1.43	7.96	363.0	2024.0	5.58
90.0	2.0	20.872	20.872	1.18	6.95	300.0	1768.0	5.89
100.0	2.0	20.888	20.888	1.05	6.43	266.0	1634.0	6.14
110.0	2.0	20.938	20.938	0.94	5.84	239.0	1485.0	6.21
120.0	2.0	21.002	21.002	0.76	5.17	192.0	1314.0	6.84
150.0	2.0	21.117	21.117	0.57	4.06	144.0	1033.0	7.17

Figure S11. Output from Python batch analysis of kinetic samples with initial conditions: [mTBD] = 0.0098 M; [EG] = 0.29 M (**Run 8**). Peak center and peak height correspond to the leftmost identified peak.



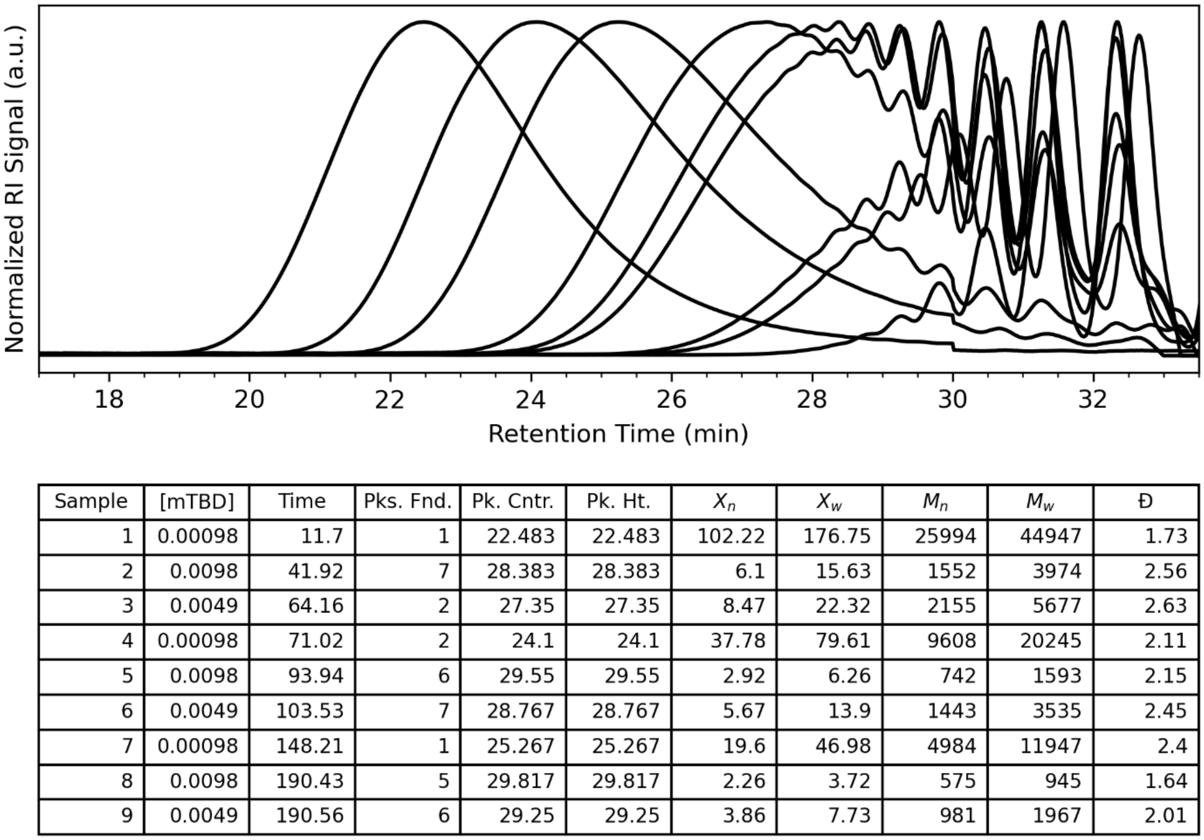
Time	Pks. Found	Pk. Center	Pk. Height	X_n	X_w	M_n	M_w	Đ
0.0	1.0	18.593	18.593	124.05	242.92	31546.0	61774.0	1.96
3.0	1.0	18.997	18.997	40.52	113.4	10303.0	28837.0	2.8
6.0	1.0	19.12	19.12	32.74	98.7	8325.0	25099.0	3.01
9.0	1.0	19.177	19.177	27.77	88.98	7062.0	22627.0	3.2
12.0	1.0	19.255	19.255	23.5	80.74	5977.0	20531.0	3.44
15.0	1.0	19.287	19.287	20.77	73.27	5283.0	18633.0	3.53
18.0	1.0	19.355	19.355	17.68	66.6	4497.0	16937.0	3.77
21.0	1.0	19.427	19.427	15.52	62.11	3947.0	15794.0	4.0
24.0	1.0	19.502	19.502	13.53	55.91	3441.0	14219.0	4.13
27.0	1.0	19.522	19.522	13.33	53.62	3390.0	13635.0	4.02
30.0	1.0	19.615	19.615	10.99	48.94	2795.0	12445.0	4.45
40.0	1.0	19.758	19.758	8.12	39.51	2064.0	10047.0	4.87
50.0	1.0	19.947	19.947	6.4	32.77	1628.0	8334.0	5.12
60.0	1.0	20.043	20.043	5.69	28.36	1446.0	7211.0	4.99
70.0	1.0	20.118	20.118	5.3	24.65	1348.0	6269.0	4.65
80.0	1.0	20.205	20.205	4.55	21.6	1158.0	5493.0	4.74
90.0	1.0	20.26	20.26	4.09	19.73	1039.0	5017.0	4.83
100.0	1.0	20.337	20.337	3.7	17.59	941.0	4473.0	4.75
110.0	1.0	20.392	20.392	3.5	16.15	891.0	4108.0	4.61
120.0	1.0	20.413	20.413	3.42	15.07	869.0	3833.0	4.41
150.0	1.0	20.56	20.56	2.72	11.99	692.0	3050.0	4.41

Figure S12. Output from Python batch analysis of kinetic samples with initial conditions: [mTBD] = 0.0049 M; [EG] = 0.193 M (**Run 9**). Peak center and peak height correspond to the leftmost identified peak.



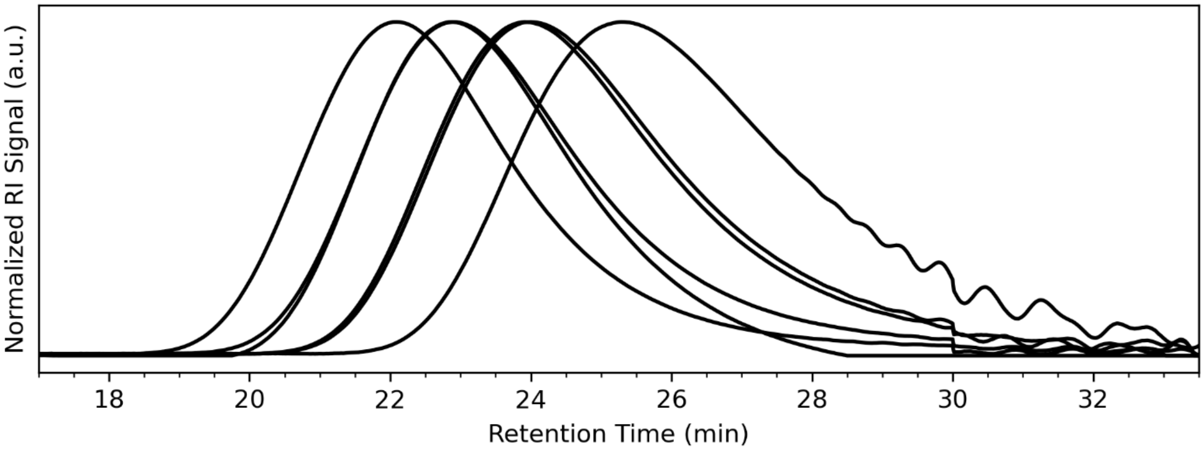
Time	Pks. Found	Pk. Center	Pk. Height	X_n	X_w	M_n	M_w	Đ
0.0	1.0	18.593	18.593	124.05	242.92	31546.0	61774.0	1.96
3.0	1.0	19.193	19.193	30.54	88.99	7767.0	22631.0	2.91
6.0	1.0	19.353	19.353	23.89	73.4	6075.0	18666.0	3.07
9.0	1.0	19.487	19.487	14.72	58.29	3743.0	14824.0	3.96
12.0	1.0	19.41	19.41	17.16	62.22	4363.0	15822.0	3.63
15.0	1.0	19.6	19.6	11.17	47.53	2840.0	12087.0	4.26
18.0	1.0	19.703	19.703	10.06	41.92	2559.0	10661.0	4.17
21.0	1.0	19.892	19.892	7.51	35.3	1910.0	8976.0	4.7
24.0	1.0	19.912	19.912	6.63	33.3	1687.0	8467.0	5.02
27.0	1.0	20.0	20.0	6.2	29.83	1577.0	7585.0	4.81
30.0	1.0	20.09	20.09	5.29	26.86	1344.0	6831.0	5.08
40.0	1.0	20.283	20.283	3.61	19.08	919.0	4852.0	5.28
50.0	1.0	20.377	20.377	3.01	16.57	765.0	4215.0	5.51
60.0	1.0	20.488	20.488	2.74	13.5	697.0	3433.0	4.93
70.0	1.0	20.538	20.538	2.39	12.26	609.0	3118.0	5.12
80.0	1.0	20.698	20.698	1.8	9.7	457.0	2466.0	5.4
90.0	1.0	20.76	20.76	1.64	8.42	417.0	2142.0	5.14
100.0	1.0	20.78	20.78	1.64	8.1	416.0	2059.0	4.95
110.0	1.0	20.86	20.86	1.29	7.12	329.0	1810.0	5.5
120.0	2.0	20.91	20.91	0.95	6.25	242.0	1590.0	6.57
150.0	2.0	21.037	21.037	0.74	5.06	189.0	1287.0	6.81

Figure S13. Output from Python batch analysis of kinetic samples with initial conditions: [mTBD] = 0.0049 M; [EG] = 0.384 M (**Run 10**). Peak center and peak height correspond to the leftmost identified peak.



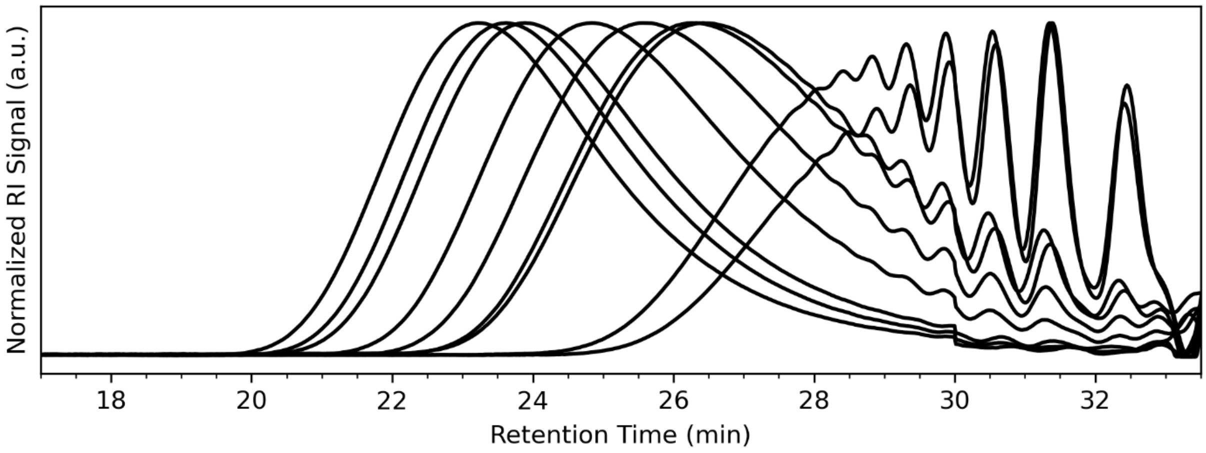
Sample	[mTBD]	Time	Pks. Fnd.	Pk. Cntr.	Pk. Ht.	X_n	X_w	M_n	M_w	Đ
1	0.00098	11.7	1	22.483	22.483	102.22	176.75	25994	44947	1.73
2	0.0098	41.92	7	28.383	28.383	6.1	15.63	1552	3974	2.56
3	0.0049	64.16	2	27.35	27.35	8.47	22.32	2155	5677	2.63
4	0.00098	71.02	2	24.1	24.1	37.78	79.61	9608	20245	2.11
5	0.0098	93.94	6	29.55	29.55	2.92	6.26	742	1593	2.15
6	0.0049	103.53	7	28.767	28.767	5.67	13.9	1443	3535	2.45
7	0.00098	148.21	1	25.267	25.267	19.6	46.98	4984	11947	2.4
8	0.0098	190.43	5	29.817	29.817	2.26	3.72	575	945	1.64
9	0.0049	190.56	6	29.25	29.25	3.86	7.73	981	1967	2.01

Figure S14. Output from Python batch analysis of first AI-guided sample collection series. The concentration of mTBD is provided in units of mol/L. Peak center and peak height correspond to the leftmost identified peak.



Sample	[mTBD]	Time	Pks. Fnd.	Pk. Cntr.	Pk. Ht.	X_n	X_w	M_n	M_w	Đ
1	0.00098	3.94	1	22.1	22.1	126.98	218.04	32292	55447	1.72
2	0.0049	3.92	1	22.9	22.9	79.7	141.58	20267	36005	1.78
3	0.0098	3.91	2	23.917	23.917	46.25	86.25	11762	21934	1.86
4	0.0049	14.33	1	24.0	24.0	42.46	82.46	10797	20969	1.94
5	0.0098	14.32	1	25.283	25.283	19.18	46.19	4877	11745	2.41
6	0.00098	24.71	1	22.883	22.883	92.57	147.27	23540	37450	1.59

Figure S15. Output from Python batch analysis of second AI-guided sample collection series. The concentration of mTBD is provided in units of mol/L. Peak center and peak height correspond to the leftmost identified peak.



Sample	[mTBD]	Time	Pks. Fnd.	Pk. Cntr.	Pk. Ht.	X_n	X_w	M_n	M_w	Đ
1	0.0049	24.68	1	24.833	24.833	25.48	56.32	6480	14323	2.21
2	0.0098	24.64	1	26.417	26.417	9.99	28.88	2540	7343	2.89
3	0.00098	35.06	1	23.217	23.217	66.79	119.76	16985	30454	1.79
4	0.0049	35.07	1	25.6	25.6	15.63	40.78	3975	10370	2.61
5	0.00098	46.24	1	23.617	23.617	54.31	99.2	13812	25227	1.83
6	0.0049	45.43	1	26.333	26.333	11.01	30.71	2800	7809	2.79
7	0.00098	55.77	1	23.883	23.883	45.89	87.16	11670	22164	1.9
8	0.0098	55.83	7	28.833	28.833	4.91	12.27	1248	3121	2.5
9	0.0098	66.19	6	29.367	29.367	4.27	10.18	1085	2589	2.39

Figure S16. Output from Python batch analysis of third AI-guided sample collection series. The concentration of mTBD is provided in units of mol/L. Peak center and peak height correspond to the leftmost identified peak.

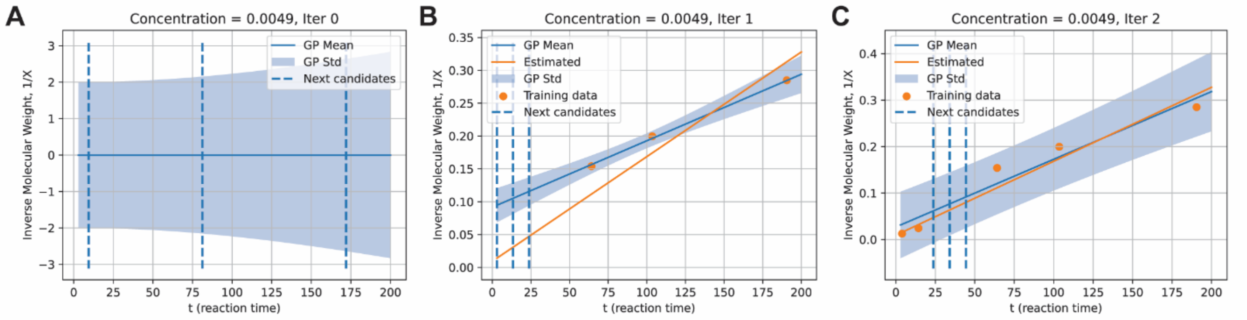


Figure S17. Iterative evolution of the surrogate model across active-learning cycles for kinetic experiments using 5 mol% mTBD (conditions from runs 1-6). The GP mean and Std represent the posterior expected prediction and associated uncertainty of the target response conditioned on the available training data. Newly acquired training data progressively refine the kinetic trend and reduce uncertainty, while the estimated regression line over the training points provides a slope-based estimate of the rate constant.

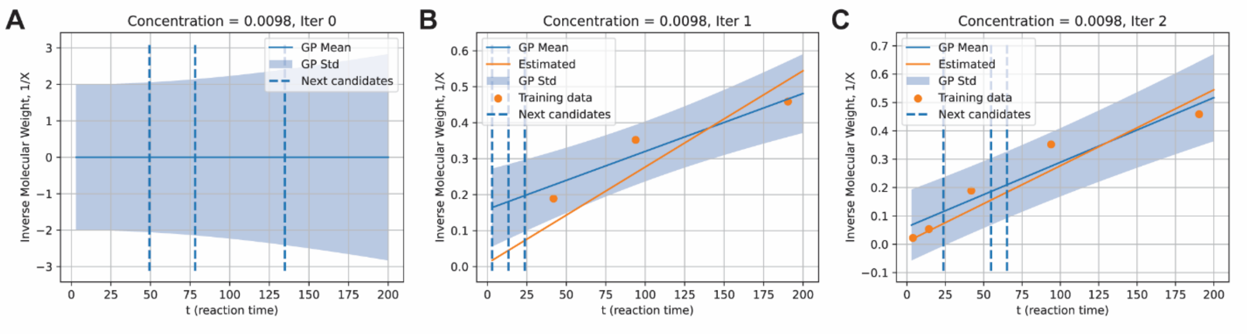


Figure S18. Iterative evolution of the surrogate model across active-learning cycles for kinetic experiments using 10 mol% mTBD (conditions from run 8). The GP mean and Std represent the posterior expected prediction and associated uncertainty of the target response conditioned on the available training data. Newly acquired training data progressively refine the kinetic trend and reduce uncertainty, while the estimated regression line over the training points provides a slope-based estimate of the rate constant.

Tables S3. ANCOVA comparing the slopes of linearized plots for automated kinetic experiments (run 7) from our initial workflow to those from AI-guided experiments. The analysis shows no significant difference in slopes.

```
                              OLS Regression Results
==============================================================================
Dep. Variable:                   Conv   R-squared:                       0.996
Model:                            OLS   Adj. R-squared:                  0.996
Method:                 Least Squares   F-statistic:                     2097.
Date:                Fri, 20 Mar 2026   Prob (F-statistic):           3.57e-28
Time:                        08:26:36   Log-Likelihood:                 165.56
No. Observations:                  27   AIC:                            -323.1
Df Residuals:                      23   BIC:                            -317.9
Df Model:                           3
Covariance Type:            nonrobust
===================================================================================
                      coef    std err          t      P>|t|      [0.025      0.975]
-----------------------------------------------------------------------------------
Intercept           0.0006      0.000      2.083      0.049    4.45e-06       0.001
Group[T.B]          0.0009      0.000      2.510      0.020       0.000       0.002
Time                0.0002   4.72e-06     49.430      0.000       0.000       0.000
Group[T.B]:Time -1.345e-05    5.9e-06     -2.281      0.032   -2.57e-05   -1.25e-06
==============================================================================
Omnibus:                       39.160   Durbin-Watson:                   1.896
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              143.222
Skew:                          -2.715   Prob(JB):                     7.94e-32
Kurtosis:                      12.890   Cond. No.                         319.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

Table S4. ANCOVA comparing the slopes of linearized plots for automated kinetic experiments (runs 1-3) from our initial workflow to those from AI-guided experiments. The analysis shows no significant difference in slopes.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                   Conv   R-squared:                       0.995
Model:                            OLS   Adj. R-squared:                  0.995
Method:                 Least Squares   F-statistic:                     3961.
Date:                Fri, 20 Mar 2026   Prob (F-statistic):           6.73e-69
Time:                        08:26:36   Log-Likelihood:                 270.98
No. Observations:                  64   AIC:                            -534.0
Df Residuals:                      60   BIC:                            -525.3
Df Model:                           3
Covariance Type:            nonrobust
===================================================================================
                      coef    std err          t      P>|t|      [0.025      0.975]
-----------------------------------------------------------------------------------
Intercept           0.0004      0.002      0.167      0.868      -0.004       0.005
Group[T.B]          0.0026      0.002      1.095      0.278      -0.002       0.007
Time                0.0013   4.38e-05     29.262      0.000       0.001       0.001
Group[T.B]:Time  8.059e-05   4.57e-05      1.765      0.083   -1.07e-05       0.000
==============================================================================
Omnibus:                        5.734   Durbin-Watson:                   0.615
Prob(Omnibus):                  0.057   Jarque-Bera (JB):                4.946
Skew:                           0.527   Prob(JB):                       0.0843
Kurtosis:                       3.862   Cond. No.                         579.
==============================================================================


Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

Table S5. ANCOVA comparing the slopes of linearized plots for automated kinetic experiments (runs 8) from our initial workflow to those from AI-guided experiments. The analysis shows no significant difference in slopes.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                   Conv   R-squared:                       0.995
Model:                            OLS   Adj. R-squared:                  0.994
Method:                 Least Squares   F-statistic:                     1330.
Date:                Fri, 20 Mar 2026   Prob (F-statistic):           5.20e-25
Time:                        08:26:36   Log-Likelihood:                 88.859
No. Observations:                  26   AIC:                            -169.7
Df Residuals:                      22   BIC:                            -164.7
Df Model:                           3
Covariance Type:            nonrobust
===================================================================================
                      coef    std err          t      P>|t|      [0.025      0.975]
-----------------------------------------------------------------------------------
Intercept          -0.0125      0.006     -2.164      0.042      -0.025      -0.001
Group[T.B]          0.0123      0.007      1.864      0.076      -0.001       0.026
Time                0.0032      0.000     29.027      0.000       0.003       0.003
Group[T.B]:Time    -0.0002      0.000     -1.872      0.075      -0.000     2.5e-05
==============================================================================
Omnibus:                        6.210   Durbin-Watson:                   1.548
Prob(Omnibus):                  0.045   Jarque-Bera (JB):                4.627
Skew:                           0.641   Prob(JB):                       0.0989
Kurtosis:                       4.621   Cond. No.                         375.
==============================================================================

Notes:
 [1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

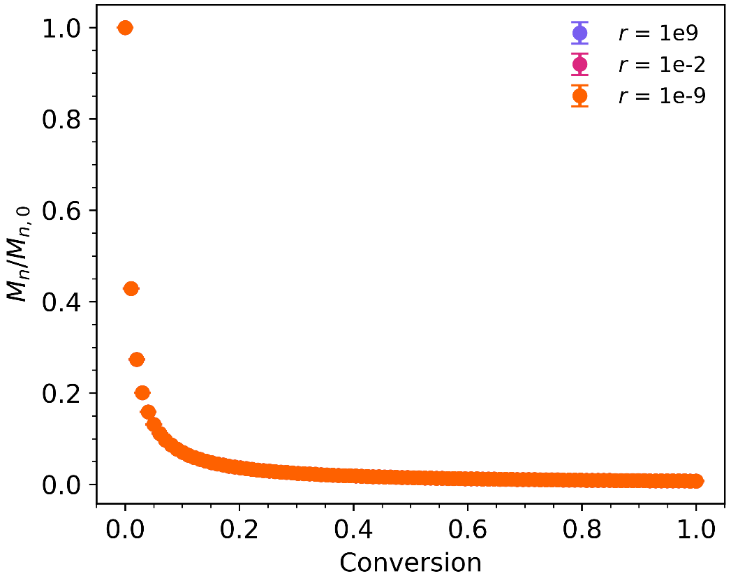


Figure S19. Overlaid M_n decay curves for simulations with different initial r.

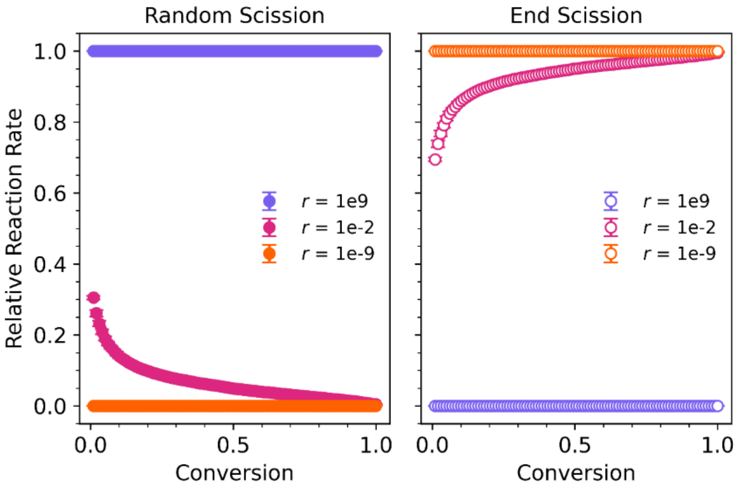


Figure S20. Relative reaction rate plots for simulations with different initial r.

End Group Analysis of Polycarbonate Oligomers by ^{1}H NMR. To determine the M_n of various PC oligomers, the integration of signal **d** corresponding to the end group aromatic protons at ~7.0 ppm was calibrated to 4.0. The integration corresponding to aromatic protons **a** and **b** from the repeat unit was then divided by 8.0, and this value was multiplied by the molar mass of the repeat unit (254.3 g/mol).

Figure S21. Representative ^{1}H NMR spectrum of a polycarbonate oligomer isolated from a 90-minute deconstruction reaction.

Determination of t-Bu Content by ^{1}H NMR. To determine the percentage of *t*-Bu chain ends present in a PC oligomer, the integration of signal **d** corresponding to the end group aromatic protons at ~7.0 ppm was calibrated to 4.0. The integration corresponding to the methyl protons **g** at 1.25 ppm was then divided by 18.0.

Figure S22. Representative ^{1}H NMR stack of *t*-Bu content determination for a series of polycarbonate oligomers.

Table S6. *t*-Bu content of twenty autonomously synthesized deconstruction oligomers as determined by ^{1}H NMR spectroscopy (see Figure S22 above).

Time (min)	***t*-Bu Content (%)**
0	100
3	37.6
6	29.4
9	26.7
12	23.2
15	20.8
18	19.9
21	19.6
24	17.1
27	16.1
30	14.5
40	12.8
50	11.4
60	10.3
70	10.2
80	8.5
90	8.44
100	8.39
110	8.11
120	7.94
150	7.61

Figure S23. *t*-Bu content of twenty autonomously synthesized PC oligomers plotted against deconstruction time.

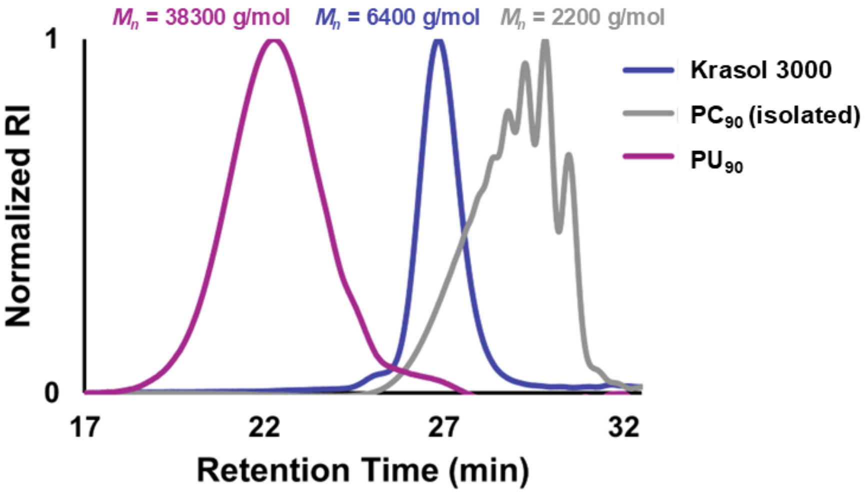


Figure S24. Overlaid SEC Chromatograms of $\mathbf{PU_{90}}$, **PC**, and **Krasol 3000**.

Determination of Polycarbonate Incorporation in PU_{90}. To determine the incorporation of PC in PU_{90}, the combined integration of the aliphatic proton signals corresponding to Krasol 3000 with chemical shift 0.50 – 1.38 ppm (highlighted in blue) was calibrated to 16.0 and then the mole fraction of PC was calculated using the integration of the repeat aromatic protons in the PC segment with chemical shift ~7.1 – 7.2 ppm (highlighted in gray).

Figure S25. ^1H NMR Spectrum of PU_{90} synthesized from a PC_{90} hard segment and Krasol 3000 soft segment.

PC_{90} Autonomous Macromonomer Purification. Solutions of polycarbonate oligomer synthesized from a 90-min automated deconstruction were combined into a round bottom flask and concentrated by evaporation. The oligomer solution was then precipitated 3x into methanol and dried under vacuum before use in polyurethane synthesis. The final M_n of the combined oligomer was 2,485 g/mol.

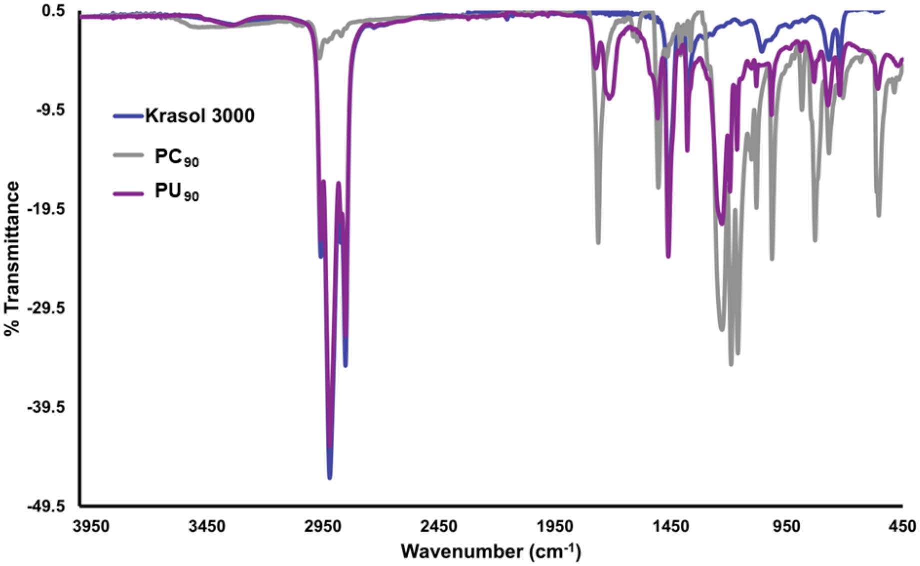


Figure S26. Overlaid IR spectra of **PC_{90}**, **Krasol 3000**, and **PU_{90}**.

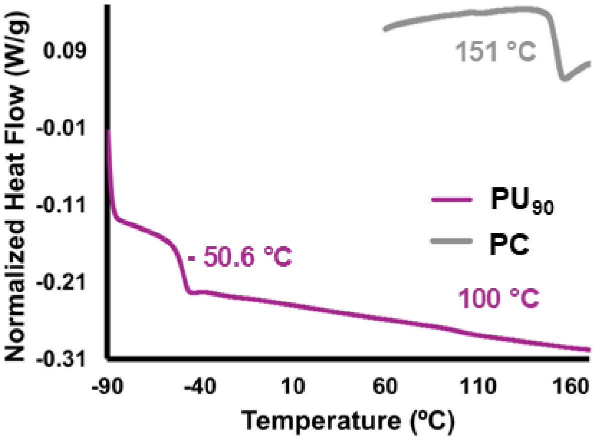


Figure S27. Overlaid DSC thermograms of **PU_{90}** and **PC**.

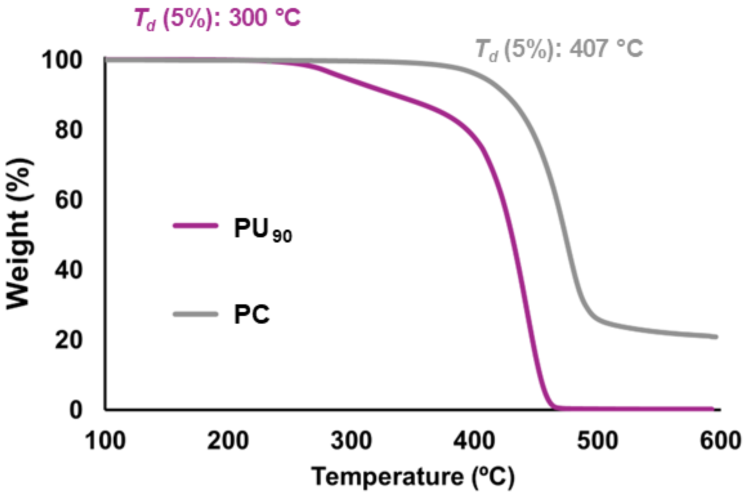


Figure S28. Overlaid TGA thermograms of **PU$_{90}$** and **PC**.

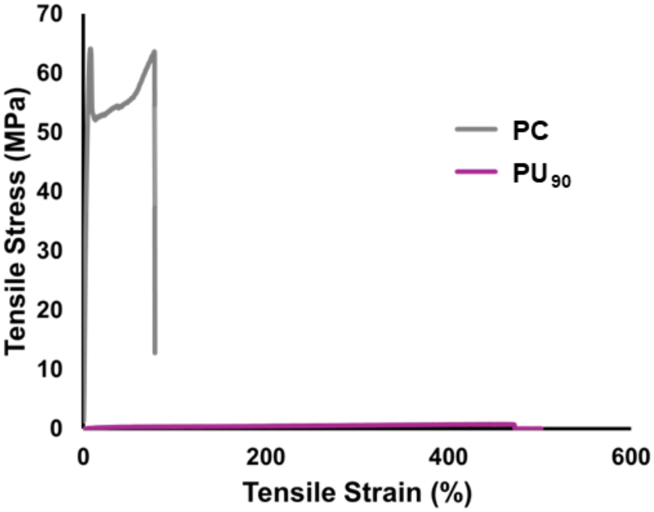


Figure S29. Overlaid Stress vs Strain plots of **PU$_{90}$** and **PC**.

Table S7. Tensile testing value for **PU$_{90}$** and **PC**. Values shown are an average of quintuplicate samples.

Polymer	Max Force [N]	Strain at break (%)	Max. Tensile Strain (mm)	Tensile Stress at Max Force (MPa)
PC	55.19 ± 5.84	81.76 ± 4.23	8.3 ± 0.43	63.45 ± 1.94
PU$_{90}$	0.410 ± 0.034	568.65 ± 52.29	64.51 ± 5.86	0.697 ± 0.037

Appendix A: Mathematical Model of Polymer Chain Scission

Experimental and simulation data are fit with a mathematical model of polymer random chain scission developed by Staggs.[5] For a population *N* polymer chains consisting of *n* repeat units and $n - 1$ bonds, let N_x represent the number of molecules in the distribution with *x* repeat units and f_x their corresponding frequency. The total number of unbroken bonds, *B*, in the distribution at conversion *p* (as defined in Eq. S6) is given by

$$B(p) = \sum_{x=1}^{n} (x - 1)\, N_x(p) \tag{S12}$$

By defining the number average degree of polymerization as

$$X_n = \sum_{x=1}^{n} x f_x(p) \tag{S13}$$

it follows that

$$1 - p = \frac{N(p)[X_n(p) - 1]}{N(0)[X_n(0) - 1]} \tag{S14}$$

where *N(0)* and *Xn(0)* represent the number of chains and their degree of polymerization, respectively, at $p = 0$. Each bond breaking event increases the total number of molecules in the population of chains such that

$$\frac{N(p)}{N(0)} = 1 + p(X_n(0) - 1) \tag{S15}$$

Combing S14 and S15 gives

$$\frac{X_n(p)}{X_n(0)} = \frac{1}{1 + p(X_n(0) - 1)} \tag{S16}$$

which can be re-arranged to linearly relate the instantaneous X_n to p

$$\frac{1}{X_n(p)} - \frac{1}{X_n(0)} = \frac{X_n(0) - 1}{X_n(0)} p \tag{S17}$$

noting that for very large polymers with $X_n(0) >> 1$ and the equation can be simplified to directly relate the inverse X_n to p. For convenience, Eq. S16 can be expressed in terms of M_w by using Eq. S5, giving the following relationship useful for fitting all forms of polymer scission data regardless of the scission mechanism

$$\frac{M_w(p)}{M_w(0)} = \frac{Đ(p)}{Đ(0)} \left[\frac{1}{1 + p(X_n(0) - 1)}\right] \tag{S18}$$

References

(1) Do, T.; Baral, E. R.; Kim, J. G. Chemical recycling of poly(bisphenol A carbonate): 1,5,7-Triazabicyclo[4.4.0]-dec-5-ene catalyzed alcoholysis for highly efficient bisphenol A and organic carbonate recovery. *Polymer* **143**, 106-114 (2018).

(2) Wang, K.-C.; Wang, S.-Y.; Kuo, C.-h.; Tseng, Y. J. Distribution-Based Classification Method for Baseline Correction of Metabolomic 1D Proton Nuclear Magnetic Resonance Spectra. *Analytical Chemistry* **85**, 1231-1239 (2013).

(3) Lynd, N. A.; Ferrier, R. C., Jr.; Beckingham, B. S. Recommendation for Accurate Experimental Determination of Reactivity Ratios in Chain Copolymerization. *Macromolecules* **52**, 2277-2285 (2019).

(4) Gillespie, D. T. Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry* **81**, 2340-2361 (1977).

(5) Staggs, J. E. J. Modelling random scission of linear polymers. *Polymer Degradation and Stability* **76**, 37-44 (2002).