AN 7–Critical Point Detection and Signal Processing

Introduction

The dielectric cure curve is characterized by four Critical Points as shown in Figure AN 7-1:

- CP(1)—A user defined level of ion viscosity that is typically used to identify the onset of material flow at the beginning of cure.

- CP(2)—Ion viscosity minimum, which typically also corresponds to the physical viscosity minimum. This Critical Point indicates the time when the crosslinking reaction and resulting increasing viscosity begin to dominate the decreasing viscosity due to melting.

- CP(3)—Inflection point, which identifies the time when the crosslinking reaction begins to slow. CP(3) is often used as a signpost that can be associated with gelation.

- CP(4)—A user defined slope that can define the end of cure. The decreasing slope corresponds to the decreasing reaction rate. Note that dielectric cure monitoring continues to reveal changes in the evolving material past the point when mechanical measurement of viscosity is not possible.
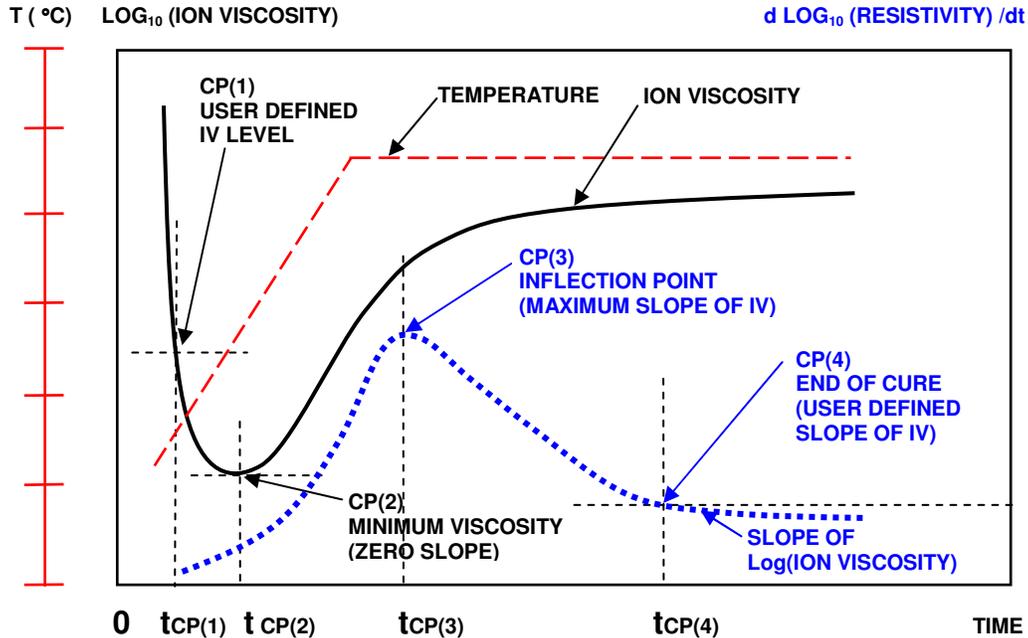

Figure AN 7-1
Ion viscosity curve and Critical Points in a curing material

In reality the data are never as smooth as shown in Figure AN 7-1, but usually have various levels of noise. Depending on the amount of noise, the Critical Points may be difficult to identify and some signal processing is almost always necessary.

The CureView™ software uses *filtering* and *averaging* with adjustable parameters allowing the user to choose the amount of processing required to discern the signal. Filtering is generally more effective than averaging at removing sharp spikes, but averaging is better at reducing background noise with a broad frequency spectrum. Optimum signal processing consists of choosing a balance between filtering and averaging to remove noise with minimal distortion of the underlying data.

Noise

Figure AN 7-2 illustrates data acquired during a bulk molding compound (BMC) cure under noisy conditions. In this case the data are displayed without any signal processing to show the raw ion viscosity and raw slope of log(ion viscosity) curves.
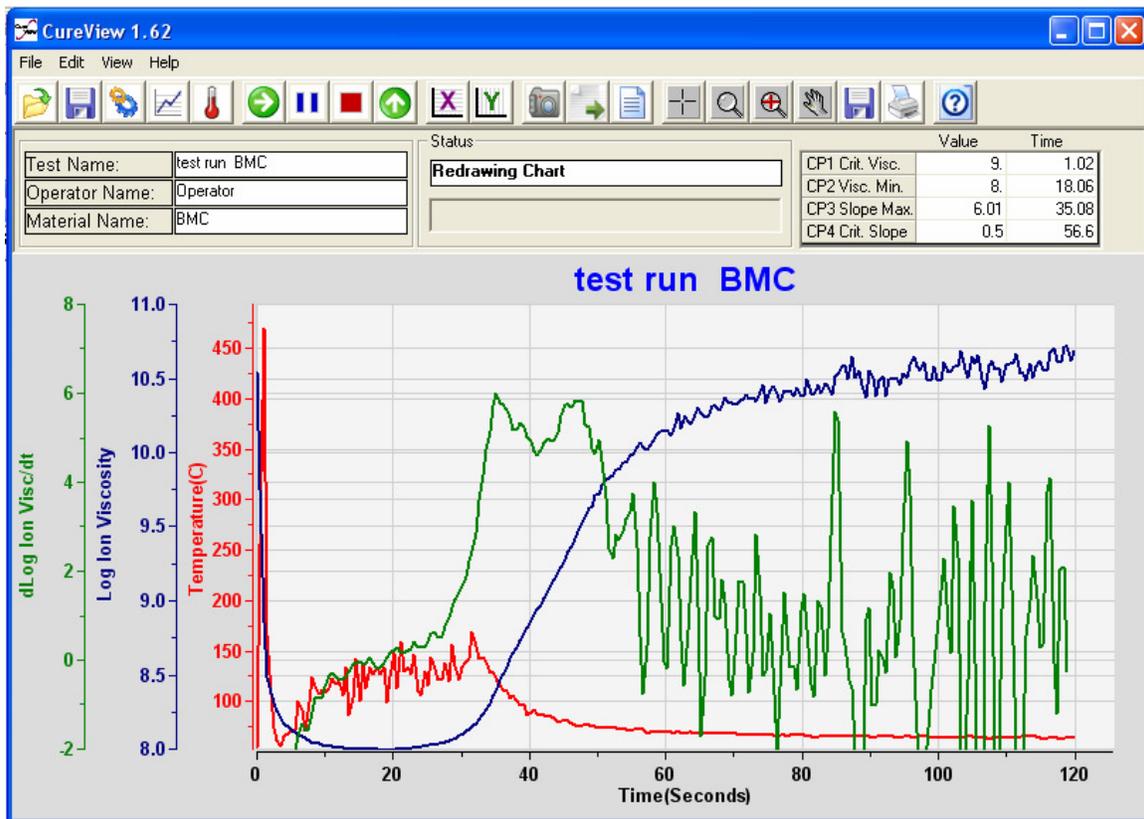

Figure AN 7-2
BMC data with no signal processing

The ion viscosity curve is smooth for the first 50 seconds. During this time the signal level is high because the material under test has low resistivity, and the signal happens to be significantly greater than the noise. But after 50 seconds the ion viscosity shows increasing noise due to reduced signal levels as the material approaches the end of cure.

The dlog(ion viscosity)/dt curve begins to exhibit noise after about 30 seconds because the slope calculation, which is a first derivative, amplifies the existing noise in the ion viscosity data. It would be very difficult to identify CP(3), the maximum slope, from this curve because of the noisy double peak between 30 and 50 seconds. It would be even more difficult to determine CP(4) because the many spikes in the data interfere with finding when the slope has reached the user defined critical value. Mathematically processing this raw data can reduce noise and yield clearer curves for ion viscosity and slope.

Filtering

CureView uses a recursive low pass filter algorithm to reduce high frequency noise in both the ion viscosity and slope of log(ion viscosity) data.

A user selected parameter in CureView determines the amount of filtering, with higher levels of filtering corresponding to lower cutoff frequencies. Increased filtering produces smoother processed data, but the data respond more slowly to change.

Filtering is performed with a time constant t_C, which is a positive number. For convenience CureView uses the following definition for the time constant:

(eq. G-1) $t_C = n * \alpha * t_M$

where n = user input integer
               α = fixed scaling factor
               t_M = measurement interval (sec)

If $t_C = 0$ then the data are not filtered.

If $t_C > 0$ then two recursion coefficients for the low pass algorithm are calculated as follows:

(eq. G-2) $A = e^{-1/tc}$

(eq. G-3) $B = 1 - A$

The input to the filtering algorithm is a series of unfiltered data points:

$d_0, d_1, d_2, d_3, \ldots d_{n-1}, d_n, d_{n+1}, \ldots$ **(unfiltered data)**

The output is a series of filtered data points:

$df_0, df_1, df_2, df_3, \ldots df_{n-1}, df_n, df_{n+1}, \ldots$ **(filtered data)**

Given unfiltered data point d_n , and the filtered data point df_{n-1}, the filtering algorithm processes d_n as follows:

(eq. G-4) $$df_n = A*df_{n-1} + B*d_n$$

The process of filtering a data series can be visualized as shown below in Figure AN 7-3:

Filtering Function

$d_0, d_1, d_2, d_3, \ldots d_{n-1},$ $d_n ,$ $d_{n+1}, d_{n+1}, d_{n+1} \ldots$
Input data series

$$A*df_{n-1} + B*d_n = df_n$$

$df_0, df_1, df_2, df_3, \ldots$ $df_{n-1},$ $df_n,$
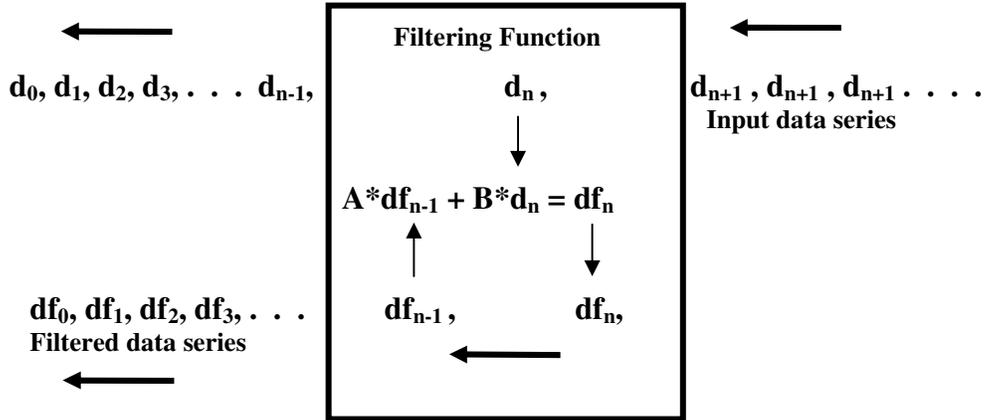**Filtered data series**

Figure AN 7-3
Generation of filtered data from raw data

Filtering can be performed on any sequence of data. Instead of filtering ion viscosity itself, CureView actually filters the gain-phase measurement data before calculating ion viscosity. CureView also filters the slope data obtained from the log(ion viscosity) curve.

CureView filters the following raw measurement data:

> Time (seconds)
> Gain (dB)
> Phase (deg)
> Temperature (°C)

The filtered data points for gain, phase and temperature are considered to occur at the time resulting from the same filtering process.

Averaging

Figure AN 7-4 shows typical raw ion viscosity data, and illustrates how CureView uses a moving boxcar, or *rolling*, average. In this example, five raw data points are averaged together to obtain each new resultant point for ion viscosity. Figure AN 7-4 also shows the relationship between the *span* and the calculation of the slope of log(ion viscosity).

Increasing the number of data points averaged together produces smoother processed data, but the result responds more slowly to change. The averaged data point is considered to occur at the time halfway between the first and last data points used in the "boxcar."
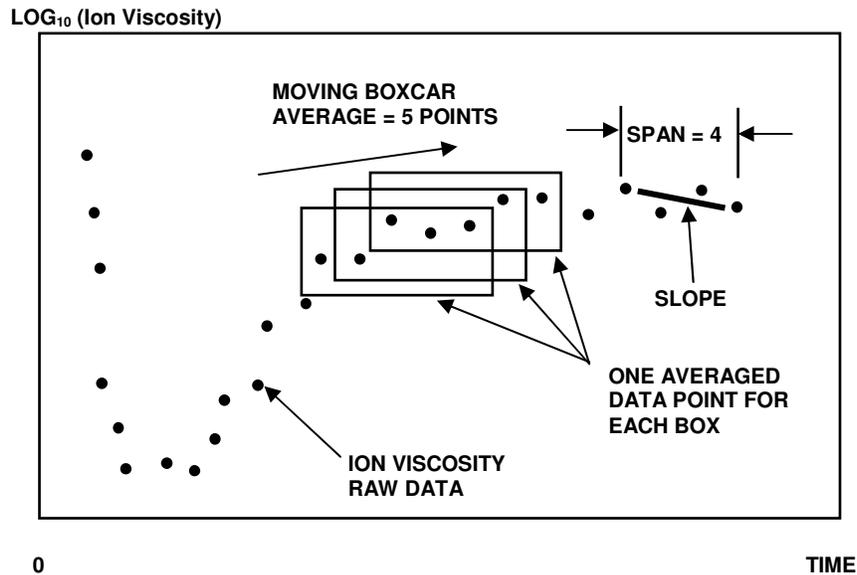
Figure AN 7-4
Signal processing: average, span and slope

While Figure AN7-4 shows averaging of the ion viscosity data as an example, averaging can be performed on any sequence of data. Instead of averaging ion viscosity itself, CureView actually averages the gain-phase measurement data before calculating ion viscosity. CureView also averages the slope data from the log(ion viscosity) curve.

CureView averages the following measurement data:

> Time (seconds)
> Gain (dB)
> Phase (deg)
> Temperature (°C)

The averaged data points for gain, phase and temperature are considered to occur at the time resulting from the same averaging process.

Span

The *span* is a parameter used to determine the slope of log(ion viscosity). As shown in Figure AN 7-4, s*pan* is the number of data points in the line segment whose slope will be calculated.

Larger values of *span* produce smoother processed data, but the result responds more slowly to change. The data point for dlog(ion viscosity)/dt is considered to occur at the time in the center of the span.

Critical Point Window

In addition to filtering and averaging data, CureView uses a window in time to restrict when it looks for each Critical Point, as shown in the example for CP(4) in Figure AN 7-5.
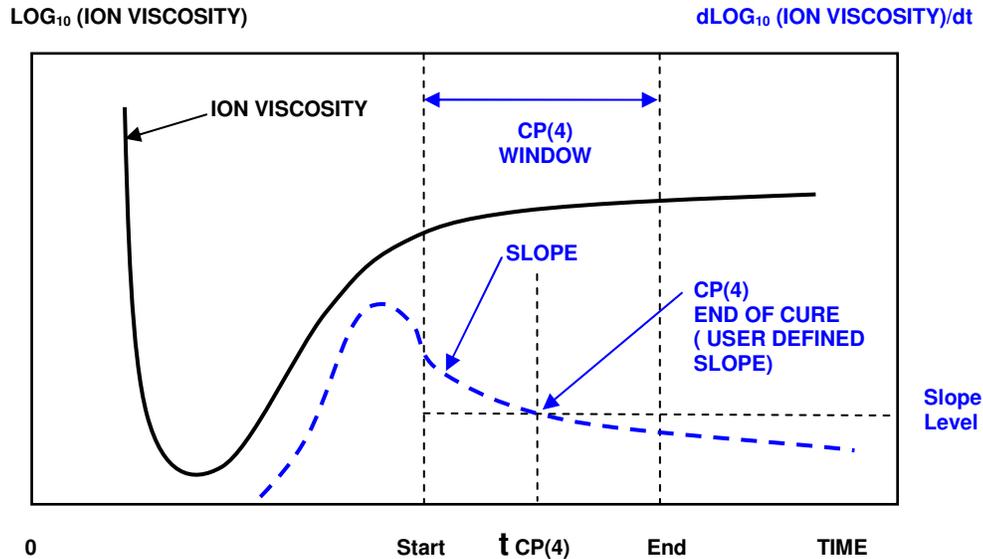
Figure AN 7-5
Window for detection of CP(4)

Good selection of window *Start* and *End* times excludes extraneous features and noise which may confuse the detection of Critical Points.

Slope Filtering

Filtering of the data for the slope of log(ion viscosity) is performed using the same function (eq. G-4) as for other data. The only difference is that all measurement data—time, gain, phase and temperature—are filtered, while slope data are filtered only after a time specified by the user.

Typically filtering for slope data is chosen to begin shortly after the expected time of the minimum ion viscosity but before the maximum of the slope of log(ion viscosity). For example, if the minimum ion viscosity of Figure AN 7-6 occurs 75 seconds after the start of cure, and if the maximum slope of log(ion viscosity) occurs at 200 seconds, then a good choice might be start slope filtering at 150 seconds. The optimum selection depends on the nature of the data and the noise, and the user may need to experiment to achieve a satisfactory result.

When the start of slope filtering is selected in this way, the filtering algorithm avoids excessive distortion of slope data in the early portion of the cure, when signal levels are high and noise is low.
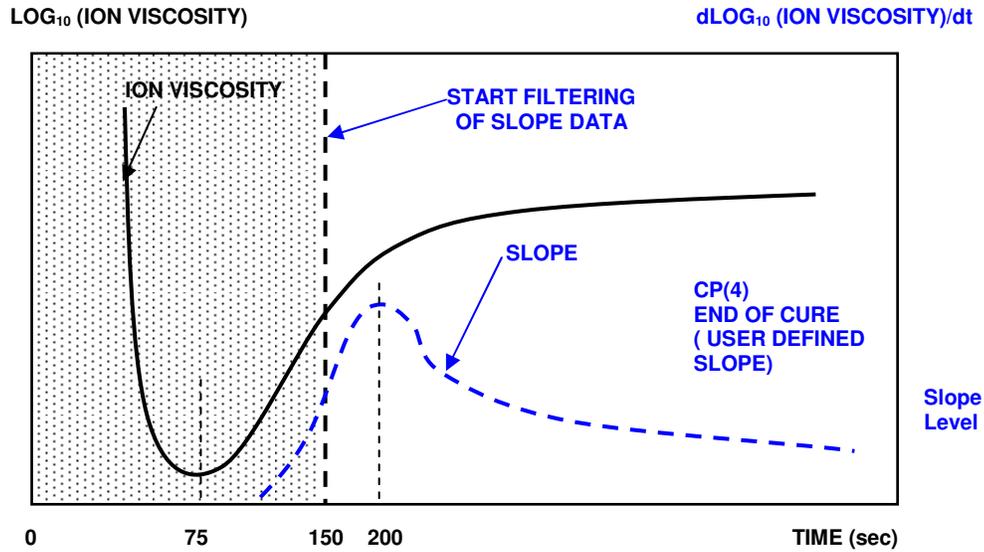
LOG₁₀ (ION VISCOSITY) — rendered as LOG_{10} (ION VISCOSITY)

dLOG₁₀ (ION VISCOSITY)/dt — rendered as dLOG_{10} (ION VISCOSITY)/dt

Figure AN 7-6
Start time for slope filtering

Results of Signal Processing

Figure AN 7-7 shows the BMC data of Figure AN 7-1 with an optimum selection of averaging and filtering parameters. Both the ion viscosity and dlog(ion viscosity)/dt curves are now smooth. The maximum slope CP(3) is clear from the single peak in the dlog(ion viscosity)/dt curve, and the critical slope for CP(4) is easy to identify.
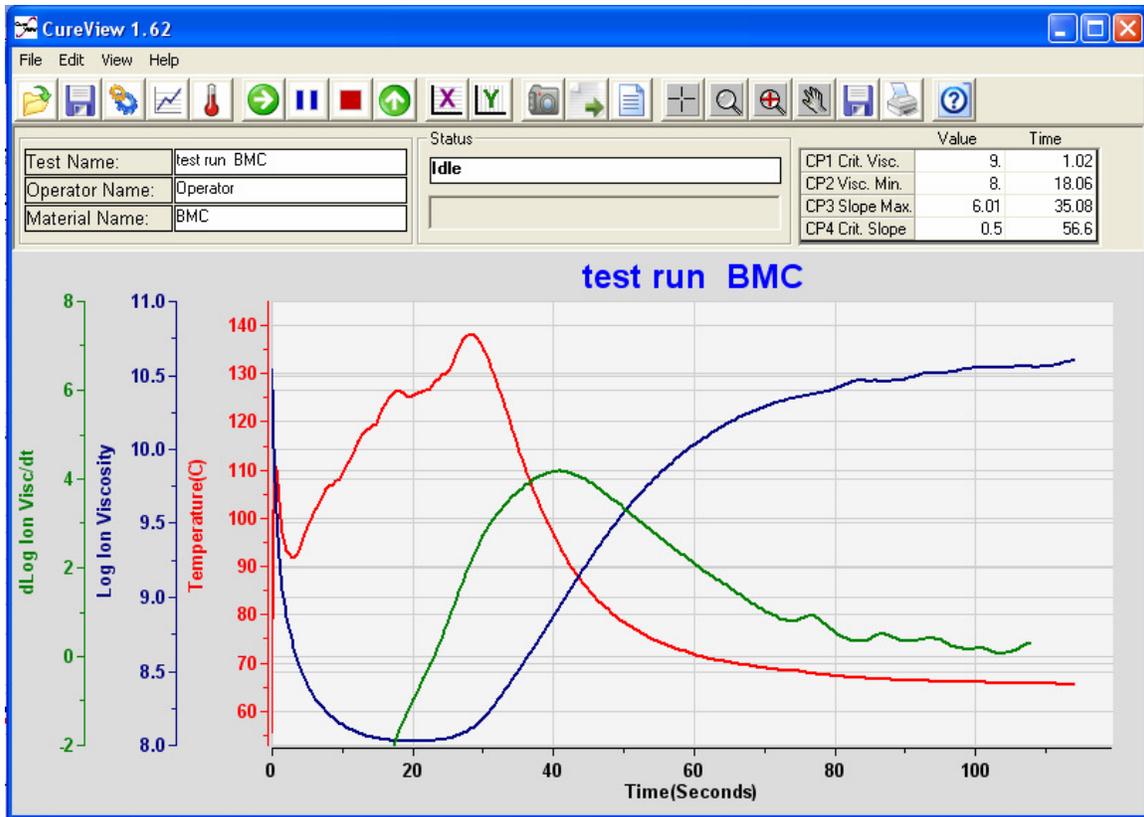
Figure AN 7-7
BMC data after averaging and filtering

Signal Processing Sequence

CureView performs signal processing of measurement data in the following order:

1. Acquire raw measurement data (time, gain, phase, temperature).

2. Filter raw measurement data (time, gain, phase, temperature).

3. Average filtered measurement data (time, gain, phase, temperature).

4. Calculate ion viscosity and slope of log(ion viscosity).

5. Filter slope of log(ion viscosity) after user selected start time for slope filter.

6. Average filtered slope data

7. Identify Critical Points (CP(1), CP(2), CP(3), CP(4)) within their respective windows.