Digital Signal Processing for Music

Part 21: Dynamics Processing

alexander lerch

dynamics processing
introduction

- **basic principle**
    - *apply time-variant audio gain*
    - gain depends on signal properties or external factors

- **applications**
    - avoid clipping (unknown input level)
    - suppress noise
    - adjust playback level (playlist)
    - decrease dynamic range (environmental noise)
    - increase loudness/energy (commercials)
    - adjust (recording) level

dynamics processing
introduction

- **basic principle**
  - *apply time-variant audio gain*
  - gain depends on signal properties or external factors

- **applications**
  - avoid clipping (unknown input level)
  - suppress noise
  - adjust playback level (playlist)
  - decrease dynamic range (environmental noise)
  - increase loudness/energy (commercials)
  - adjust (recording) level

dynamics processing
introduction: effects

- (noise) **gate**
  - suppression of low levels in pauses
- **compressor**
  - reduction of the dynamic range
- **expander**
  - expansion of the dynamic range
- **limiter**
  - limitation of maximum gain
- **AGC** (automatic gain control)
  - slow adaptation of recording/payback gain

dynamics processing
introduction: effects

- (noise) **gate**
  - suppression of low levels in pauses
- **compressor**
  - reduction of the dynamic range
- **expander**
  - expansion of the dynamic range
- **limiter**
  - limitation of maximum gain
- **AGC** (automatic gain control)
  - slow adaptation of recording/payback gain

dynamics processing
introduction: effects

- (noise) **gate**
  - suppression of low levels in pauses
- **compressor**
  - reduction of the dynamic range
- **expander**
  - expansion of the dynamic range
- **limiter**
  - limitation of maximum gain
- **AGC** (automatic gain control)
  - slow adaptation of recording/payback gain

dynamics processing
introduction: effects

- (noise) **gate**
  - suppression of low levels in pauses
- **compressor**
  - reduction of the dynamic range
- **expander**
  - expansion of the dynamic range
- **limiter**
  - limitation of maximum gain
- **AGC** (automatic gain control)
  - slow adaptation of recording/payback gain

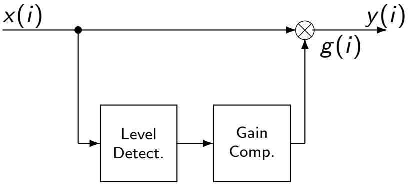
dynamics processing

introduction: effects

- (noise) **gate**
  - suppression of low levels in pauses
- **compressor**
  - reduction of the dynamic range
- **expander**
  - expansion of the dynamic range
- **limiter**
  - limitation of maximum gain
- **AGC** (automatic gain control)
  - slow adaptation of recording/payback gain

dynamics processing

overview

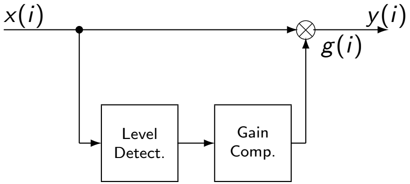


$$y(i) = x(i) \cdot g(i)$$

computation of $g(i)$ usually depends on

1. input signal *level*
2. properties & characteristics of the dynamics processor
3. time-based control mechanism

dynamics processing
overview



$$y(i) = x(i) \cdot g(i)$$

computation of $g(i)$ usually depends on

1. input signal *level*
2. properties & characteristics of the dynamics processor
3. time-based control mechanism

dynamics processing
level detection

- typical measures
  - **peak**:
    physical measure of maximum amplitude
  - **rms**:
    physical measure of power level
  - **loudness model**:
    models of loudness perception (dBA, Zwicker, BS.1770)

- **level computation**

$$v_{\mathrm{dB}}(i) = 20 \cdot \log_{10}\left(\frac{v(i)}{v_0}\right)$$

  - v_0: reference constant (0 dB point)
    digital: $v_0 = 1 \Rightarrow \mathrm{dBFS}$
  - scaling factor: 1 dB $\approx$ JNDL

dynamics processing
level detection

- typical measures
  - **peak**:
    physical measure of maximum amplitude
  - **rms**:
    physical measure of power level
  - **loudness model**:
    models of loudness perception (dBA, Zwicker, BS.1770)

- **level computation**

$$v_{\mathrm{dB}}(i) = 20 \cdot \log_{10}\left(\frac{v(i)}{v_0}\right)$$

  - v_0: reference constant (0 dB point)
    digital: $v_0 = 1 \Rightarrow \mathrm{dBFS}$
  - scaling factor: 1 dB ≈ JNDL

dynamics processing

level detection

- typical measures
  - **peak**:
    physical measure of maximum amplitude
  - **rms**:
    physical measure of power level
  - **loudness model**:
    models of loudness perception (dBA, Zwicker, BS.1770)

- **level computation**

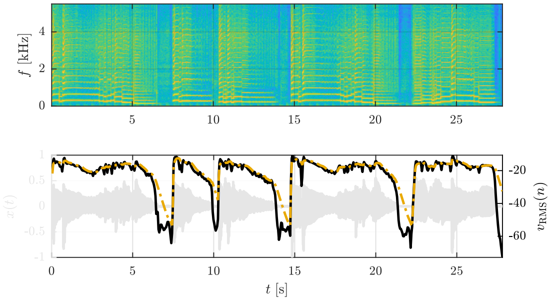
$$v_{\mathrm{dB}}(i) = 20 \cdot \log_{10}\left(\frac{v(i)}{v_0}\right)$$

  - v_0: reference constant (0 dB point)
    digital: $v_0 = 1 \Rightarrow \mathrm{dBFS}$
  - scaling factor: 1 dB $\approx$ JNDL

dynamics processing

level detection: root mean square 1/2

$$v_{\mathrm{RMS}}(n) = \sqrt{\frac{1}{\mathcal{K}} \sum_{i=i_{\mathrm{s}}(n)}^{i_{\mathrm{e}}(n)} x(i)^2}$$



matlab source: plotRms.m

dynamics processing

level detection: root mean square 2/2

sample-by-sample processing:

- reduce computational complexity

$$
\begin{aligned}
v_{\mathrm{RMS}}^2(n) &= \frac{x(i_\mathrm{e}(n))^2 - x(i_\mathrm{s}(n-1))^2}{i_\mathrm{e}(n) - i_\mathrm{s}(n) + 1} + v_{\mathrm{RMS}}^2(n-1) \\
v_{\mathrm{RMS}}(n) &= \sqrt{v_{\mathrm{RMS}}^2(n)}
\end{aligned}
$$

- single pole approximation

$$
\begin{aligned}
v_{\mathrm{tmp}}(i) &= \alpha \cdot v_{\mathrm{tmp}}(i-1) + (1-\alpha) \cdot x(i)^2 \\
v_{\mathrm{RMS}}^*(i) &= \sqrt{v_{\mathrm{tmp}}(i)}
\end{aligned}
$$

dynamics processing

level detection: root mean square 2/2

sample-by-sample processing:

- reduce computational complexity

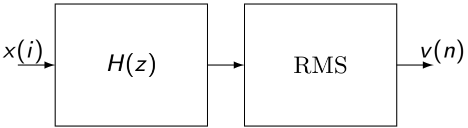
$$\begin{aligned} v^2_{\mathrm{RMS}}(n) &= \frac{x(i_{\mathrm{e}}(n))^2 - x(i_{\mathrm{s}}(n-1))^2}{i_{\mathrm{e}}(n) - i_{\mathrm{s}}(n) + 1} + v^2_{\mathrm{RMS}}(n-1) \\ v_{\mathrm{RMS}}(n) &= \sqrt{v^2_{\mathrm{RMS}}(n)} \end{aligned}$$

- single pole approximation

$$\begin{aligned} v_{\mathrm{tmp}}(i) &= \alpha \cdot v_{\mathrm{tmp}}(i-1) + (1-\alpha) \cdot x(i)^2 \\ v^*_{\mathrm{RMS}}(i) &= \sqrt{v_{\mathrm{tmp}}(i)} \end{aligned}$$

dynamics processing

level detection: weighted root mean square

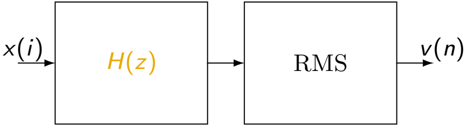


$H(z)$:

- A, B, C weighting
- RLB (BS.1770)
- ...

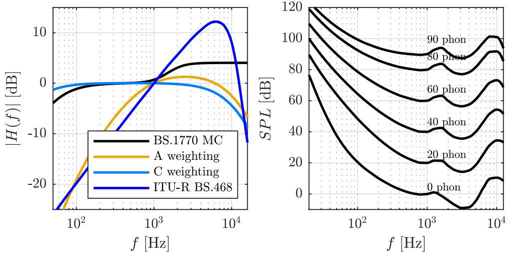
dynamics processing

level detection: weighted root mean square



$H(z)$:

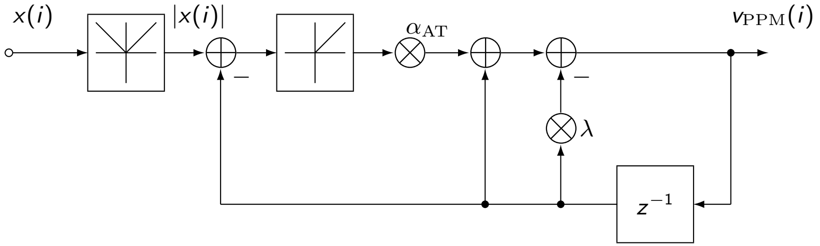
- A, B, C weighting
- RLB (BS.1770)
- ...



matlab source: plotLoudnessWeighting.m

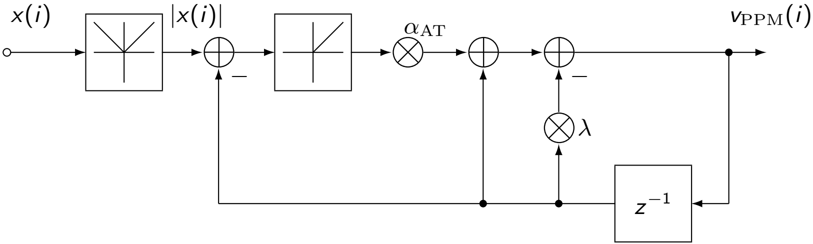
dynamics processing

level detection: peak detection (PPM) 1/2



dynamics processing

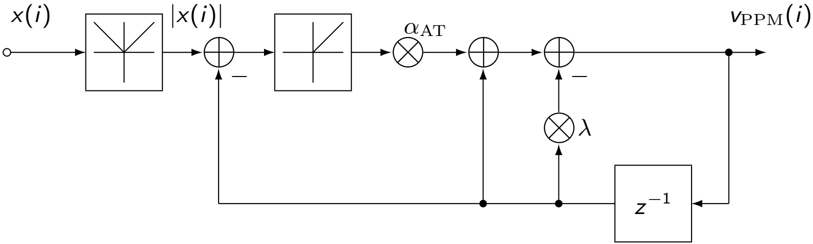
level detection: peak detection (PPM) 1/2



- **release state** $(|x(i)| \leq v_{\mathrm{PPM}}(i-1) \Rightarrow \lambda = \alpha_{\mathrm{RT}})$

dynamics processing

level detection: peak detection (PPM) 1/2

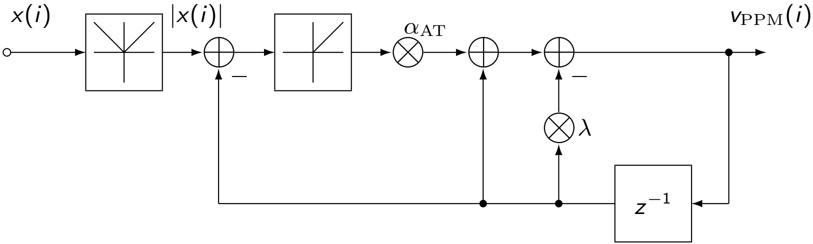


- **release state** $(|x(i)| \leq v_{\mathrm{PPM}}(i-1) \Rightarrow \lambda = \alpha_{\mathrm{RT}})$

$$\begin{aligned} v_{\mathrm{PPM}}(i) &= v_{\mathrm{PPM}}(i-1) - \alpha_{\mathrm{RT}} \cdot v_{\mathrm{PPM}}(i-1) \\ &= (1-\alpha_{\mathrm{RT}}) \cdot v_{\mathrm{PPM}}(i-1) \end{aligned}$$

dynamics processing

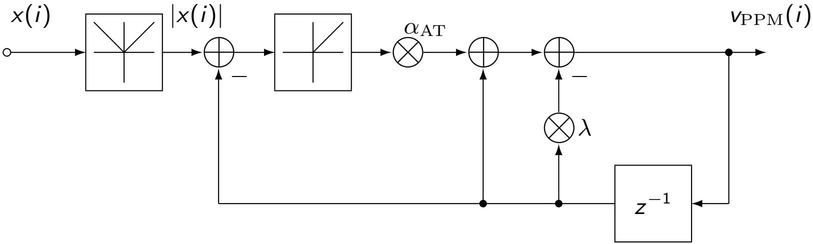
level detection: peak detection (PPM) 1/2



- **attack state** $(|x(i)| > v_{\mathrm{PPM}}(i-1) \Rightarrow \lambda = 0)$

dynamics processing

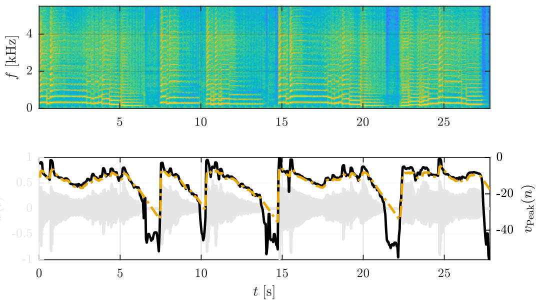
level detection: peak detection (PPM) 1/2



- **attack state** $(|x(i)| > v_{\mathrm{PPM}}(i-1) \Rightarrow \lambda = 0)$

$$\begin{aligned} v_{\mathrm{PPM}}(i) &= \alpha_{\mathrm{AT}} \cdot \left(|x(i)| - v_{\mathrm{PPM}}(i-1)\right) + v_{\mathrm{PPM}}(i-1) \\ &= \alpha_{\mathrm{AT}} \cdot |x(i)| + (1 - \alpha_{\mathrm{AT}}) \cdot v_{\mathrm{PPM}}(i-1) \end{aligned}$$

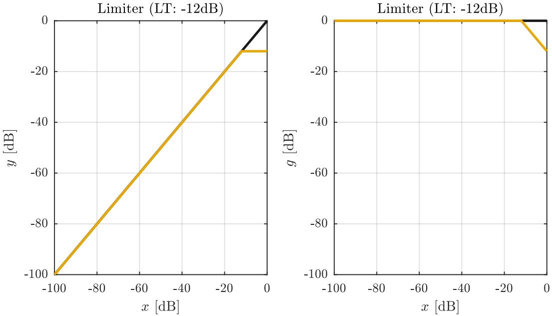
dynamics processing
level detection: peak detection (PPM) 2/2



matlab source: plotTimePeakEnvelope.m

dynamics processing

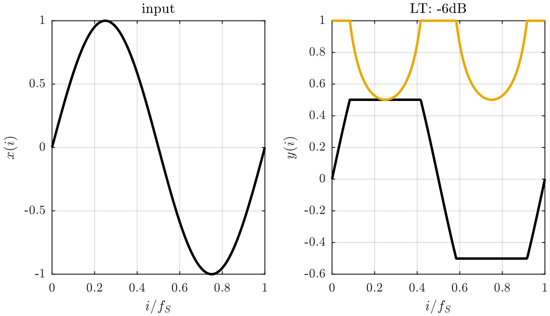
response curve: limiter



matlab source: plotDynamicsResponseCurve-1.m

dynamics processing

response curve: limiter

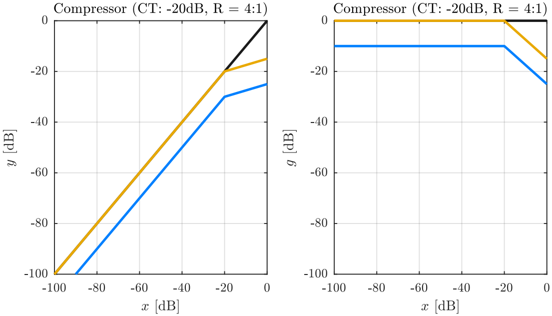


param $LT = -9\,\mathrm{dB}$ w/o gain smoothing:

matlab source: plotDynamics-1.m

dynamics processing

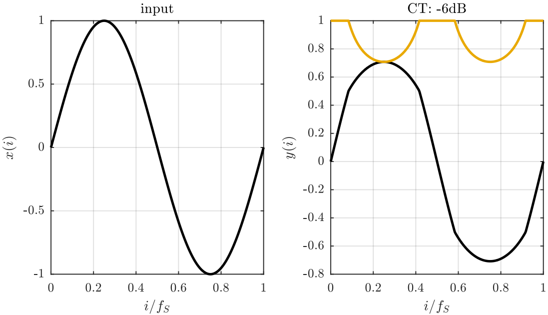
response curve: compressor



matlab source: plotDynamicsResponseCurve-2.m

dynamics processing

response curve: compressor

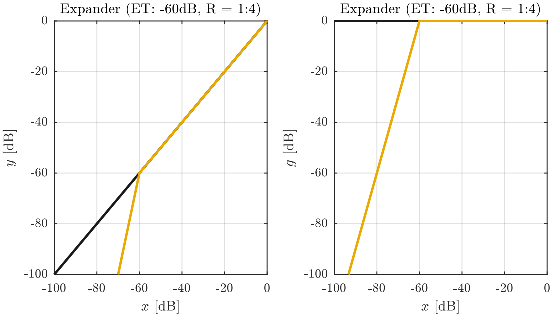


param $CT = -9\,\mathrm{dB}$ w/o gain smoothing:

matlab source: plotDynamics-2.m

dynamics processing

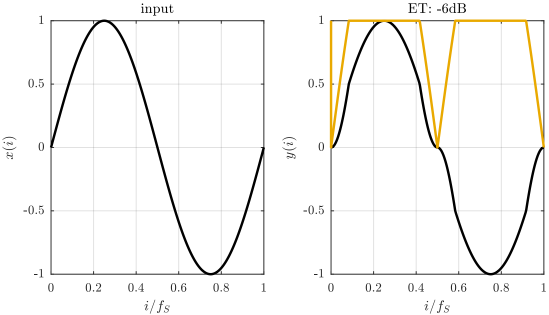
response curve: expander



matlab source: plotDynamicsResponseCurve-3.m

dynamics processing

response curve: expander

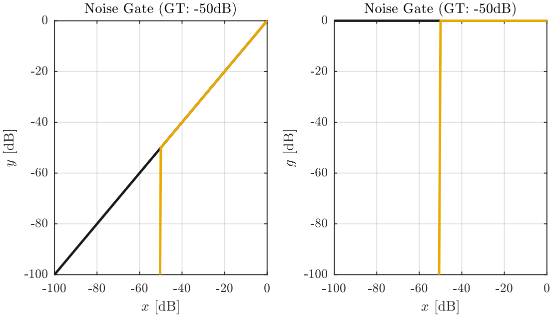


param $ET = -6\,\mathrm{dB}$ w/o gain smoothing:

matlab source: plotDynamics-3.m

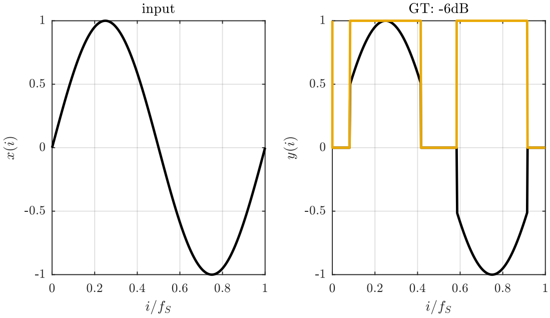
dynamics processing

response curve: noise gate



matlab source: plotDynamicsResponseCurve-4.m

dynamics processing
response curve: noise gate



param $NT = -12\,\mathrm{dB}$ w/o gain smoothing:

matlab source: plotDynamics-4.m

dynamics processing

response curve: mathematical description (compressor)

logarithmic description, nonlinear part

- **output**: $Y = g(X) + X \quad \mathrm{[dB]}$
- **ratio**: $R = \frac{\Delta L_i}{\Delta L_o}$
- **slope**: $CS = 1 - \frac{1}{R}$
- **linear equation** (offset CT): $Y = \frac{1}{R}(X - CT) + CT$
- **gain** ($g = Y - X$):

$$
\begin{aligned}
g &= \frac{1}{R}(X - CT) + CT - X \\
&= \left(1 - \frac{1}{R}\right) \cdot (CT - X) \\
&= CS \cdot (CT - X)
\end{aligned}
$$

dynamics processing

response curve: mathematical description (compressor)

logarithmic description, nonlinear part

- **output**: $Y = g(X) + X \quad [\mathrm{dB}]$
- **ratio**: $R = \frac{\Delta L_i}{\Delta L_o}$
- **slope**: $CS = 1 - \frac{1}{R}$
- **linear equation** (offset CT): $Y = \frac{1}{R}(X - CT) + CT$
- **gain** ($g = Y - X$):

$$\begin{aligned} g &= \frac{1}{R}(X - CT) + CT - X \\ &= \left(1 - \frac{1}{R}\right) \cdot (CT - X) \\ &= CS \cdot (CT - X) \end{aligned}$$

dynamics processing

response curve: mathematical description (compressor)

logarithmic description, nonlinear part

- **output**: $Y = g(X) + X \quad [\mathrm{dB}]$
- **ratio**: $R = \frac{\Delta L_i}{\Delta L_o}$
- **slope**: $CS = 1 - \frac{1}{R}$
- **linear equation** (offset CT): $Y = \frac{1}{R}(X - CT) + CT$
- **gain** ($g = Y - X$):

$$\begin{aligned} g &= \frac{1}{R}(X - CT) + CT - X \\ &= \left(1 - \frac{1}{R}\right) \cdot (CT - X) \\ &= CS \cdot (CT - X) \end{aligned}$$

dynamics processing

response curve: mathematical description (compressor)

logarithmic description, nonlinear part

- **output**: $Y = g(X) + X \quad [\mathrm{dB}]$
- **ratio**: $R = \frac{\Delta L_i}{\Delta L_o}$
- **slope**: $CS = 1 - \frac{1}{R}$
- **linear equation** (offset CT): $Y = \frac{1}{R}(X - CT) + CT$
- **gain** ($g = Y - X$):

$$\begin{aligned} g &= \frac{1}{R}(X - CT) + CT - X \\ &= \left(1 - \frac{1}{R}\right) \cdot (CT - X) \\ &= CS \cdot (CT - X) \end{aligned}$$

dynamics processing

response curve: mathematical description (compressor)

logarithmic description, nonlinear part

- **output**: $Y = g(X) + X \quad [\mathrm{dB}]$
- **ratio**: $R = \frac{\Delta L_i}{\Delta L_o}$
- **slope**: $CS = 1 - \frac{1}{R}$
- **linear equation** (offset CT): $Y = \frac{1}{R}(X - CT) + CT$
- **gain** ($g = Y - X$):

$$
\begin{aligned}
g &= \frac{1}{R}(X - CT) + CT - X \\
&= \left(1 - \frac{1}{R}\right) \cdot (CT - X) \\
&= CS \cdot (CT - X)
\end{aligned}
$$

dynamics processing

response curve: mathematical description (compressor)

logarithmic description, nonlinear part

- **output**: $Y = g(X) + X \quad [\mathrm{dB}]$
- **ratio**: $R = \frac{\Delta L_i}{\Delta L_o}$
- **slope**: $CS = 1 - \frac{1}{R}$
- **linear equation** (offset CT): $Y = \frac{1}{R}(X - CT) + CT$
- **gain** ($g = Y - X$):

$$
\begin{aligned}
g &= \frac{1}{R}(X - CT) + CT - X \\
&= \left(1 - \frac{1}{R}\right) \cdot (CT - X) \\
&= CS \cdot (CT - X)
\end{aligned}
$$

dynamics processing

response curve: mathematical description (summary 1/2)

logarithmic description, nonlinear part

- **limiter**

$$\begin{aligned} R &= \infty \\ Y &= LT \\ g &= LT - X \end{aligned}$$

- **compressor**

$$\begin{aligned} R &> 1 \\ Y &= \frac{1}{R}(X - CT) + CT \\ g &= \left(1 - \frac{1}{R}\right) \cdot (CT - X) \end{aligned}$$

dynamics processing

response curve: mathematical description (summary 1/2)

logarithmic description, nonlinear part

- **limiter**

$$\begin{aligned} R &= \infty \\ Y &= LT \\ g &= LT - X \end{aligned}$$

- **compressor**

$$\begin{aligned} R &> 1 \\ Y &= \frac{1}{R}(X - CT) + CT \\ g &= \left(1 - \frac{1}{R}\right) \cdot (CT - X) \end{aligned}$$

dynamics processing

response curve: mathematical description (summary 2/2)

logarithmic description, nonlinear part

- **expander**

$$
\begin{aligned}
R &< 1 \\
Y &= \frac{1}{R}(X - ET) + ET \\
g &= \left(1 - \frac{1}{R}\right) \cdot (ET - X)
\end{aligned}
$$

- **gate**

$$
\begin{aligned}
R &= 0 \\
Y &= -\infty \\
g &= -\infty
\end{aligned}
$$

dynamics processing

response curve: mathematical description (summary 2/2)

logarithmic description, nonlinear part

- **expander**

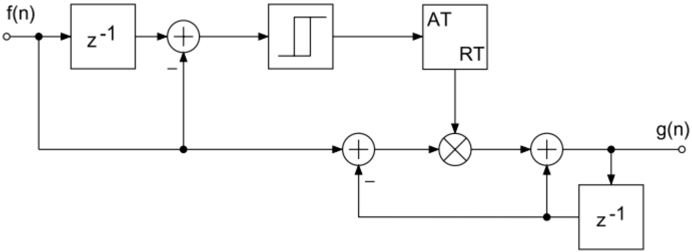
$$\begin{aligned} R &< 1 \\ Y &= \frac{1}{R}(X - ET) + ET \\ g &= \left(1 - \frac{1}{R}\right) \cdot (ET - X) \end{aligned}$$

- **gate**

$$\begin{aligned} R &= 0 \\ Y &= -\infty \\ g &= -\infty \end{aligned}$$

dynamics processing

smoothing: attack and release 1/2

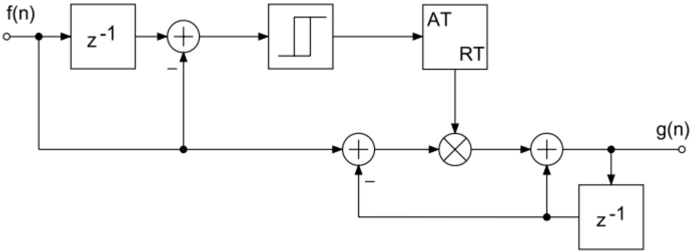


- α_{AT}: attack constant
- α_{RT}: release constant

$$
\begin{aligned}
g(n) &= \alpha \cdot (f(n) - g(n-1)) + g(n-1) \\
&= \alpha f(n) + (1-\alpha) \cdot g(n-1)
\end{aligned}
$$

dynamics processing

smoothing: attack and release 1/2

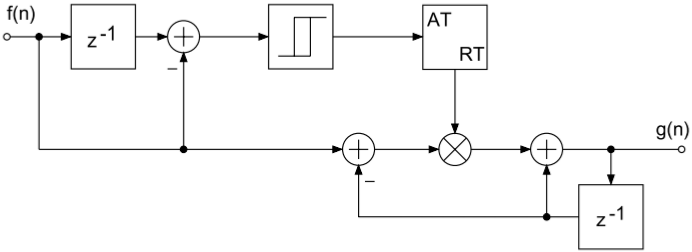


- α_{AT}: attack constant
- α_{RT}: release constant

$$\begin{aligned} g(n) &= \alpha \cdot (f(n) - g(n-1)) + g(n-1) \\ &= \alpha f(n) + (1-\alpha) \cdot g(n-1) \end{aligned}$$

dynamics processing

smoothing: attack and release 1/2

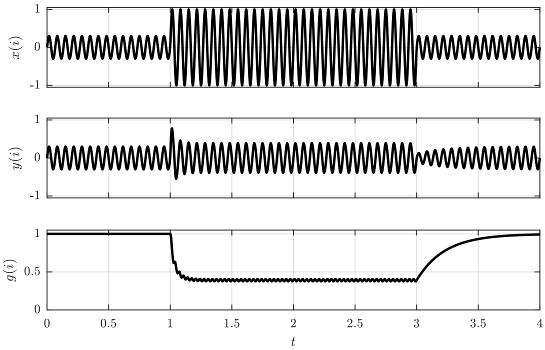


- α_{AT}: attack constant
- α_{RT}: release constant

$$
\begin{aligned}
g(n) &= \alpha \cdot (f(n) - g(n-1)) + g(n-1) \\
&= \alpha f(n) + (1-\alpha) \cdot g(n-1)
\end{aligned}
$$

dynamics processing

smoothing: attack and release 2/2



matlab source: plotAttackRelease.m

dynamics processing

smoothing: attack and release coefficients

- single pole step response → $g(t) = 1 - e^{\frac{-t}{\tau}}$
- define single pole integration time between 10% and 90%

$$
\begin{aligned}
t_\mathrm{I} &= t_{90} - t_{10} \\
0.1 &= 1 - e^{\frac{-t_{10}}{\tau}} \\
0.9 &= 1 - e^{\frac{-t_{90}}{\tau}} \\
\Rightarrow {}^{0.9}\!/_{0.1} &= e^{\frac{t_{90}-t_{10}}{\tau}} \\
\log\left({}^{0.9}\!/_{0.1}\right) &= {}^{t_{90}-t_{10}}\!/_{\tau} \\
t_{90} - t_{10} &= 2.197\tau \\
\tau &\approx {}^{t_\mathrm{I}}\!/_{2.2}
\end{aligned}
$$

dynamics processing

smoothing: attack and release coefficients

- single pole step response → $g(t) = 1 - e^{\frac{-t}{\tau}}$
- define single pole integration time between 10% and 90%

$$\begin{aligned}
t_\mathrm{I} &= t_{90} - t_{10}\\
0.1 &= 1 - e^{\frac{-t_{10}}{\tau}}\\
0.9 &= 1 - e^{\frac{-t_{90}}{\tau}}\\
\Rightarrow {}^{0.9}/_{0.1} &= e^{\frac{t_{90}-t_{10}}{\tau}}\\
\log\left({}^{0.9}/_{0.1}\right) &= {}^{t_{90}-t_{10}}/_{\tau}\\
t_{90} - t_{10} &= 2.197\tau\\
\tau &\approx {}^{t_\mathrm{I}}/_{2.2}
\end{aligned}$$

dynamics processing

smoothing: attack and release coefficients

- single pole step response $\rightarrow g(t) = 1 - e^{\frac{-t}{\tau}}$
- define single pole integration time between 10% and 90%

$$\begin{aligned}
t_\mathrm{I} &= t_{90} - t_{10} \\
0.1 &= 1 - e^{\frac{-t_{10}}{\tau}} \\
0.9 &= 1 - e^{\frac{-t_{90}}{\tau}} \\
\Rightarrow {}^{0.9}/_{0.1} &= e^{\frac{t_{90}-t_{10}}{\tau}} \\
\log\left({}^{0.9}/_{0.1}\right) &= {}^{t_{90}-t_{10}}/_{\tau} \\
t_{90} - t_{10} &= 2.197\tau \\
\tau &\approx {}^{t_\mathrm{I}}/_{2.2}
\end{aligned}$$

dynamics processing

smoothing: attack and release coefficients

- single pole step response $\rightarrow g(t) = 1 - e^{\frac{-t}{\tau}}$
- define single pole integration time between 10% and 90%

$$\begin{aligned}
t_{\mathrm{I}} &= t_{90} - t_{10} \\
0.1 &= 1 - e^{\frac{-t_{10}}{\tau}} \\
0.9 &= 1 - e^{\frac{-t_{90}}{\tau}} \\
\Rightarrow {}^{0.9}\!/\!{}_{0.1} &= e^{\frac{t_{90}-t_{10}}{\tau}} \\
\log\left({}^{0.9}\!/\!{}_{0.1}\right) &= {}^{t_{90}-t_{10}}\!/\!{}_{\tau} \\
t_{90} - t_{10} &= 2.197\tau \\
\tau &\approx {}^{t_{\mathrm{I}}}\!/\!{}_{2.2}
\end{aligned}$$

dynamics processing

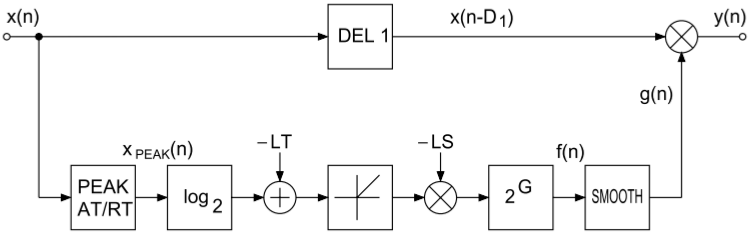
smoothing: attack and release coefficients

- single pole step response $\rightarrow g(t) = 1 - e^{\frac{-t}{\tau}}$
- define single pole integration time between 10% and 90%

$$
\begin{aligned}
t_\mathrm{I} &= t_{90} - t_{10} \\
0.1 &= 1 - e^{\frac{-t_{10}}{\tau}} \\
0.9 &= 1 - e^{\frac{-t_{90}}{\tau}} \\
\Rightarrow {}^{0.9}\!/_{0.1} &= e^{\frac{t_{90}-t_{10}}{\tau}} \\
\log\left({}^{0.9}\!/_{0.1}\right) &= {}^{t_{90}-t_{10}}\!/_{\tau} \\
t_{90} - t_{10} &= 2.197\tau \\
\tau &\approx {}^{t_\mathrm{I}}\!/_{2.2}
\end{aligned}
$$

dynamics processing

overall system: limiter

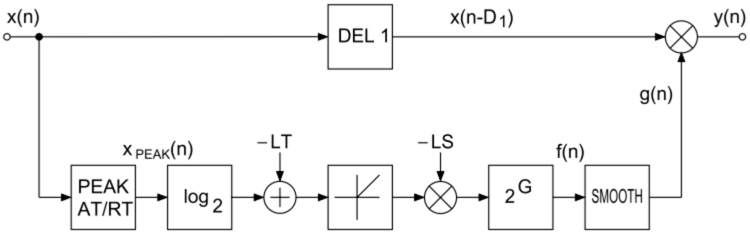


$$CS = 1 - \frac{1}{R} \Rightarrow LS = 1$$

- $X < LT \rightarrow g = 1$
- $X > LT \rightarrow g = (LT - X)$

dynamics processing

overall system: limiter

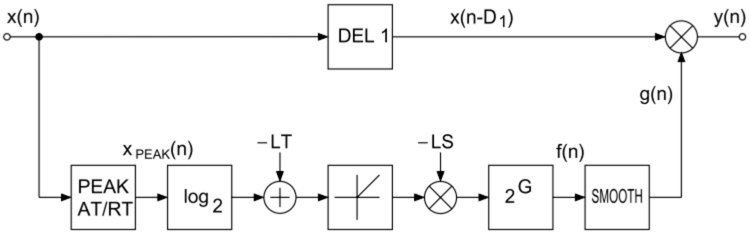


$$CS = 1 - \frac{1}{R} \Rightarrow LS = 1$$

- $X < LT \rightarrow g = 1$
- $X > LT \rightarrow g = (LT - X)$

dynamics processing

overall system: limiter

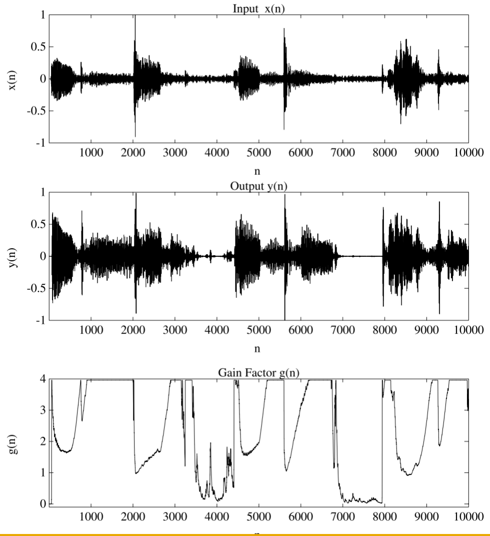


$$CS = 1 - \frac{1}{R} \Rightarrow LS = 1$$

- $X < LT \rightarrow g = 1$
- $X > LT \rightarrow g = (LT - X)$

dynamics processing

gain visualization: combined system



dynamics processing

audio examples

- Gate
- Expander
- Compressor
- Limiter

dynamics processing

variants 1/3

- **attack & release constant selection**
  - depending on "abruptness" of change
- **hold time**
  - before release, hold gain constant (avoid pumping with low frequency signals)
- **oversampling**
  - high time resolution for peak detection

matlab source: plotHiddenClipping.m

dynamics processing

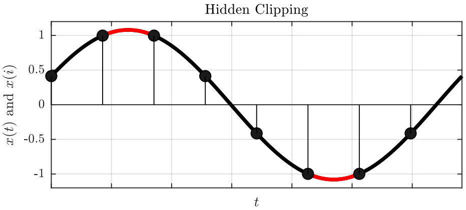
variants 1/3

- **attack & release constant selection**
  - depending on "abruptness" of change
- **hold time**
  - before release, hold gain constant (avoid pumping with low frequency signals)
- **oversampling**
  - high time resolution for peak detection

matlab source: plotHiddenClipping.m

dynamics processing
variants 1/3

- **attack & release constant selection**
  - depending on "abruptness" of change
- **hold time**
  - before release, hold gain constant (avoid pumping with low frequency signals)
- **oversampling**
  - high time resolution for peak detection



matlab source: plotHiddenClipping.m

dynamics processing

variants 2/3

- **stereo link**
  - consider both channels (avoid level-dependent changes of stereo image)
    - one master channel (left or right)
    - mean of both channels
    - channel with higher level (max)

- **soft knee**
  - smooth crossover from linear area to compressed area

  potentially noticeable with
    - very short attack times
    - high compression ratios

dynamics processing

variants 2/3

- **stereo link**
  - consider both channels (avoid level-dependent changes of stereo image)
    - one master channel (left or right)
    - mean of both channels
    - channel with higher level (max)

- **soft knee**
  - smooth crossover from linear area to compressed area

  potentially noticeable with
    - very short attack times
    - high compression ratios

dynamics processing
variants 2/3

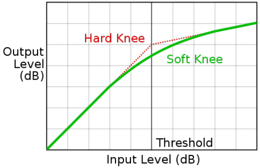
- **stereo link**
  - consider both channels (avoid level-dependent changes of stereo image)
    - one master channel (left or right)
    - mean of both channels
    - channel with higher level (max)
- **soft knee**
  - smooth crossover from linear area to compressed area

  potentially noticeable with
    - very short attack times
    - high compression ratios

dynamics processing
variants 2/3

- **stereo link**
  - consider both channels (avoid level-dependent changes of stereo image)
    - one master channel (left or right)
    - mean of both channels
    - channel with higher level (max)
- **soft knee**
  - smooth crossover from linear area to compressed area

    potentially noticeable with
    - very short attack times
    - high compression ratios



https://en.wikipedia.org/wiki/File:Compression_knee.svg

dynamics processing

variants 3/3

- **side chain**
  - choose different input signal for level control ("ducking")
- **look-ahead**
  - introduce higher delay in signal path
    - shift gain modification in time
    - combine "future" measurement with current
- **multi-band compression**
  - apply one compressor to each frequency band
  - advantages:
    - avoid pumping: varying level in one band (e.g. bass drum) does not influence gain of other bands
    - maximize power, overall loudness

dynamics processing
variants 3/3

- **side chain**
  - choose different input signal for level control ("ducking")
- **look-ahead**
  - introduce higher delay in signal path
    - shift gain modification in time
    - combine "future" measurement with current
- **multi-band compression**
  - apply one compressor to each frequency band
  - advantages:
    - avoid pumping: varying level in one band (e.g. bass drum) does not influence gain of other bands
    - maximize power, overall loudness

dynamics processing

variants 3/3

- **side chain**
  - choose different input signal for level control ("ducking")
- **look-ahead**
  - introduce higher delay in signal path
    - shift gain modification in time
    - combine "future" measurement with current
- **multi-band compression**
  - apply one compressor to each frequency band
  - advantages:
    - avoid pumping: varying level in one band (e.g. bass drum) does not influence gain of other bands
    - maximize power, overall loudness

dynamics processing

parameter ranges

- **threshold**
  $-120 \ldots 0$ dB
- **ratio**
  $0.05 \ldots 20$ (Limiter: ∞)
- **attack**
  $0 \ldots 10$ ms
- **release**
  $20 \ldots 300$ ms
- **hold**
  $0 \ldots 10$ ms
- **stereo-link**
  On/Off
- **oversampling**
  $1 \ldots 8$
- **look-ahead**
  $0 \ldots 500$ ms

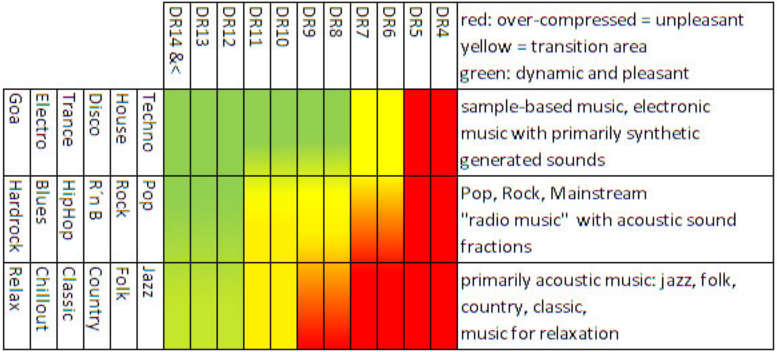
dynamics processing

parameter ranges

- **threshold**
  $-120 \dots 0$ dB
- **ratio**
  $0.05 \dots 20$ (Limiter: ∞)
- **attack**
  $0 \dots 10$ ms
- **release**
  $20 \dots 300$ ms
- **hold**
  $0 \dots 10$ ms
- **stereo-link**
  On/Off
- **oversampling**
  $1 \dots 8$
- **look-ahead**
  $0 \dots 500$ ms

dynamics processing

dynamic range target (opinionated)



somewhere on the internet

dynamics processing
summary

dynamics processing systems are

- **time variant**:
  gain changes over time
- **signal adaptive**:
  gain depends on (input) signal
- sometimes **non-linear**:
  at very short attack times (limiting)