Scaling Definitions

1) Consensus (Old)

Define

T_R = 3000 bytes ; Reference Transaction weight for the current Monero implementation

Z_M = 300000 bytes ; Minimum penalty free zone

M_B = Block weight in bytes

M_L = the median over the last 100000 blocks of max((min (M_B , 1.4M_L), Z_M) ; recursive calculation for M_L with M_L starting at Z_M ; Effective long term median

Z_E = Z_M ; Effective penalty free zone

M_S = the median over the last 100 blocks of max(M_B , Z_E) ; Effective short term median.

M_N = min(M_S , 50M_L) ; Median for penalty calculation.

R_{Base} = Block Reward

$0 < M_B \leq 2M_N$; Requirement for valid block

$B = M_B / M_N - 1$ where $-1 < B \leq 1$

$P_B = R_{Base}B^2$ for $B > 0$; Monero applies a penalty P_B, to increase the block weight by B.

$P_B = 0$ for $B \leq 0$

$dP_B / dB = 2R_{Base}B$

Proposed Scaling Definitions (October 2020)

1) Consensus (New)

Define

T_R = 2000 bytes ; Reference Transaction weight for the upcoming, in October 2020, Monero fork.

Z_M = 300000 bytes ; Minimum penalty free zone

M_B = Block weight in bytes

M_L = The median over the last 100000 blocks of max((min (M_B , 2M_L), Z_M , M_L/2) ; recursive calculation for M_L with M_L starting at Z_M ; Effective long term median

Z_E = max (Z_M , M_L/4) ; Effective penalty free zone

M_S = the median over the last 100 blocks of max(min(M_B , 50M_L) , Z_E) ; Effective short term median

M_N = M_S ; Median for penalty calculation.

R_{Base} = Block Reward

$0 < M_B \leq 2M_N$; Requirement for valid block

$B = M_B / M_N - 1$ where $-1 < B \leq 1$

$P_B = R_{Base}B^2$ for $B > 0$; Monero applies a penalty P_B, to increase the block weight by B.

$P_B = 0$ for $B \leq 0$

$dP_B / dB = 2R_{Base}B$

For Subsequent forks after CLSAG (October 2020)

T_{RC} = Weight in bytes of a 2 input and 2 output transaction rounded up to the nearest 100 bytes. If a future Monero fork does not permit a 2 input and 2 output transaction then the permitted transaction with the lowest number of inputs and outputs greater than or equal to 2 is used instead.

T_{RP} = T_R of previous Monero version

Z_{MP} = Z_M of previous Monero version

For: $T_{RC} > T_{RP}$ Then
$Z_M \geq Z_{MP}T_{RC}/T_{RP}$, $T_R = T_{RC}$; $Z_M > Z_{MP}T_{RC}/T_{RP}$ may, for example, be chosen in order to keep the total fee T_R the same as T_{RP}.

For $T_{RC} = T_{RP}$ Then
$Z_M = Z_{MP}$, $T_R = T_{RC}$

For $T_{RC} < T_{RP}$ Then
$Z_M = Z_{MP}$, $T_R = T_{RC}$ or for a small change $T_R = T_{RP}$

2) Minimum Fee For Node Relay (Old)

We add a, penalty attracting, transaction T with a size of T_T to a block of weight M_B

Define

$B_T = T_T / M_N$

$P_{BT} = R_{Base}(B+B_T)^2 = R_{base}(B^2 + 2BB_T + B_T^2)$; The new penalty, where $B + B_T > 0$

$P_T = P_{BT} – P_B = R_{Base}(2BB_T + B_T^2)$; Increase in penalty from adding transaction T

$F_T = R_{Base}(2BB_T + B_T^2)$; The additional fee required to overcome the increase in penalty P_T

For the case B = 0 this reduces to $F_T = R_{Base}B_T^2$

To calculate the minimum fee we consider a transaction of weight T_R at the start of the penalty, B = 0 with $M_N = Z_M$. 20% of fee required to pay the penalty incurred is the minimum fee.

$M_F = \min(M_N , M_L)$; Median for minimum fee calculation

$B_R = T_R / Z_M$

$F_R = R_{Base}B_R^2$; Fee required to pay the penalty incurred

$f_R = R_{Base}B_R/M_F$; Fee required to pay the penalty incurred fee per byte for a given M_F

$f_M = 0.2f_R$; Minimum fee per byte

Examples:

T_R = 3000 bytes, R_{Base} = 1.5 XMR

M_N = 300000 bytes, M_L = 300000 bytes
$Z_E = Z_M$ = 300000 bytes
$f_M = 0.2*R_{base}B_R/M_F$ = 10.0 nXMR / byte

M_N = 30000000 bytes, M_L = 30000000 bytes
$Z_E = Z_M$ = 300000 bytes
$f_M = 0.2*R_{base}B_R/M_F$ = 0.100 nXMR / byte

2) Minimum Fee For Node Relay (New)

We add a, penalty attracting, transaction T with a size of T_T to a block of weight M_B

Define

$B_T = T_T / M_N$

$P_{BT} = R_{Base}(B+B_T)^2 = R_{base}(B^2 + 2BB_T + B_T^2)$; The new penalty, where $B + B_T > 0$

$P_T = P_{BT} – P_B = R_{Base}(2BB_T + B_T^2)$; Increase in penalty from adding transaction T

$F_T = R_{Base}(2BB_T + B_T^2)$; The additional fee required to overcome the increase in penalty P_T

For the case B = 0 this reduces to $F_T = R_{Base}B_T^2$

To calculate the minimum fee we consider a transaction of weight T_R at the start of the penalty, B = 0 with $M_N = Z_M$. The fee required to pay the penalty incurred is the minimum fee.

$M_F = \min(M_N , M_L)$; Median for minimum fee calculation

$B_R = T_R / Z_M$

$F_R = R_{Base}B_R^2$; Fee required to pay the penalty incurred

$f_R = R_{Base}B_R/M_F$; Fee required to pay the penalty incurred per byte for a given M_F

$f_M = f_R$; Minimum fee per byte

Examples:

T_R = 2000 bytes, R_{Base} = 1.5 XMR

M_N = 300000 bytes, M_L = 300000 bytes
$Z_E = Z_M$ = 300000 bytes
$f_M = R_{base}B_R/M_F$ = 33.3 nXMR / byte

M_N = 30000000 bytes, M_L = 30000000 bytes
Z_E = 7500000 bytes, Z_M = 300000 bytes
$f_M = R_{base}B_R/M_F$ = 0.333 nXMR / byte

T_R = 2700 bytes, R_{Base} = 1.5 XMR

M_N = 300000 bytes, M_L = 300000 bytes
$Z_E = Z_M$ = 300000 bytes
$f_M = R_{base}B_R/M_F$ = 45.0 nXMR / byte

M_N = 30000000 bytes, M_L = 30000000 bytes
Z_E = 7500000 bytes, Z_M = 300000 bytes
$f_M = R_{base}B_R/M_F$ = 0.450 nXMR / byte

3) Wallet Fees (Old)

For the calculation of wallet fees we assume that the next 10 blocks have,no transactions other than the coinbase transaction, the empty blocks. We then calculate M_{SW} by following the calculation of M_S at this future point. We use the previous 10000 blocks and the future 10 empty blocks.

Define

M_{BW} = Block weight in bytes for past and current real and future empty blocks

M_{SW} = the median over the “last” 100 blocks of max(M_{BW} , Z_M) ;

$M_{NW} = \min(M_{SW} , 50M_L)$

$M_{FW} = \min(M_{NW} , M_L)$; Median for wallet fee calculation

$F_N = R_{Base}B_R^2$; Normal Transaction fee at minimum fee and minimum $M_{FW} = Z_M$

$f_N = R_{Base}B_R/M_{FW}$; Normal Transaction fee per byte for a given M_{FW}

$f_L = 0.2f_N$;Low Transaction fee per byte for a given M_{FW}

$f_M = 5f_N$;High Transaction fee per byte for a given M_{FW}

$f_P = 2R_{Base}/M_{NW} = 2f_NM_{FW}/(B_RM_{NW})$; Maximum Penalty (B =1) Transaction fee per byte for a given M_{NW}

$f_H = f_P$; Highest Transaction fee per byte.

3) Wallet Fees (New)

For the calculation of wallet fees we assume that the next 10 blocks have,no transactions other than the coinbase transaction, the empty blocks. We then calculate M_{LW} and M_{SW} by following the calculation of M_L and M_S at this future point. We use the previous 100000 blocks and the future 10 empty blocks.

Define

M_{BW} = Block weight in bytes for past and current real and future empty blocks

M_{LW} = The median over the “last” 100000 blocks of max((min (M_{BW} , $2M_L$), Z_M , $M_{LW}/2$) ; recursive calculation for M_{LW} over the “next” 10 blocks with M_{LW} starting at M_L ; Effective long term median for wallet fees

Z_{EW} = max (Z_M , $M_{LW}/4$) ; Effective penalty free zone for wallet fees cannot fall below $M_{LW}/4$, Z_M

M_{SW} = the median over the “last” 100 blocks of max(min(M_{BW} , $50M_{LW}$) , Z_{EW}) ; Effective short term median for wallet fees

$M_{NW} = M_{SW}$

$M_{FW} = \min(M_{NW} , M_{LW})$; Median for wallet fee calculation

$T_{RW} = 1.1T_R$; Reference transaction for lowest (normal) wallet fee calculation ; A 10% safety margin is added at this point.

$B_{RW} = T_{RW} / Z_M$

$F_N = R_{Base}B_{RW}^2$; Lowest (Normal) Transaction fee at minimum fee and minimum $M_{FW} = Z_M$

$f_N = R_{Base}B_{RW}/M_{FW}$; Lowest (Normal) Transaction fee per byte for a given M_{FW}

$f_L = 4f_N$;Low Transaction fee per byte for a given M_{FW}

$f_M = 16f_N$;Medium Transaction fee per byte for a given M_{FW}

$f_P = 2R_{Base}/M_{NW} = 2f_NM_{FW}/(B_{RW}M_{NW})$; Maximum Penalty (B =1) Transaction fee per byte for a given M_{NW}

$f_H = 64f_N\max(1, M_{FW}/(32B_{RW}M_{NW}))$; High Transaction fee per byte

Wallet Fee Rounding

Wallet fees, f_N, f_L, f_M, and f_H are rounded up to the desired number of significant digits in the significant

Wallet Fee Rounding Examples

Three significant digits

27810	Rounded to : 27900
37.94	Rounded to : 38.0
0.5555	Rounded to : 0.556
0.002342	Rounded to : 0.00235

Two significant digits

27810	Rounded to : 28000
37.94	Rounded to : 38
0.5555	Rounded to : 0.56
0.002342	Rounded to : 0.0024

Wallet Fee Examples

T_R = 3000 bytes, R_{Base} = 1.5 XMR

M_{NW} = 300000 bytes, M_L = 300000 bytes
$f_N = R_{Base}B_R/M_{FW}$ = 50.0 nXMR / byte
$f_L = 0.2f_N$ = 10.0 nXMR / byte
$f_M = 5f_N$ = 250 nXMR / byte
$f_P = 2R_{Base}/M_{NW} = 2f_NM_{FW}/(B_RM_{NW})$ = 10000 nXMR / byte
$f_H = f_P$ = 10000 nXMR / byte

M_{NW} = 15000000 bytes, M_L = 300000 bytes
$f_N = R_{Base}B_R/M_{FW}$ = 50.0 nXMR / byte
$f_L = 0.2f_N$ = 10.0 nXMR / byte
$f_M = 5f_N$ = 250 nXMR / byte
$f_P = 2R_{Base}/M_{NW} = 2f_NM_{FW}/(B_RM_{NW})$ = 200 nXMR / byte
$f_H = f_P$ = 200 nXMR / byte

Wallet Fee Examples

T_{RW} = 2200 bytes, R_{Base} = 1.5 XMR

M_{NW} = 300000 bytes, M_{LW} = 300000 bytes
$f_N = R_{Base}B_{RW}/M_{FW}$ = 36.7 nXMR / byte
$f_L = 4f_N$ = 147 nXMR / byte
$f_M = 16f_N$ = 587 nXMR / byte
$f_P = 2R_{Base}/M_{NW} = 2f_NM_{FW}/(B_{RW}M_{NW})$ = 10000 nXMR / byte
$f_H = 64f_N\max(1, M_{FW}/(32B_{RW}M_{NW}))$ = 10000 nXMR / byte

M_{NW} = 15000000 bytes, M_{LW} = 300000 bytes
$f_N = R_{Base}B_{RW}/M_{FW}$ = 36.7 nXMR / byte
$f_L = 4f_N$ = 147 nXMR / byte
$f_M = 16f_N$ = 587 nXMR / byte
$f_P = 2R_{Base}/M_{NW} = 2f_NM_{FW}/(B_{RW}M_{NW})$ = 200 nXMR / byte
$f_H = 64f_N\max(1, M_{FW}/(32B_{RW}M_{NW}))$ = 2350 nXMR / byte

T_{RW} = 2970 bytes, R_{Base} = 1.5 XMR

M_{NW} = 300000 bytes, M_{LW} = 300000 bytes
$f_N = R_{Base}B_{RW}/M_{FW}$ = 49.5 nXMR / byte
$f_L = 4f_N$ = 198 nXMR / byte
$f_M = 16f_N$ = 792 nXMR / byte
$f_P = 2R_{Base}/M_{NW} = 2f_NM_{FW}/(B_{RW}M_{NW})$ = 10000 nXMR / byte
$f_H = 64f_N\max(1, M_{FW}/(32B_{RW}M_{NW}))$ = 10000 nXMR / byte

M_{NW} = 15000000 bytes, M_{LW} = 300000 bytes
$f_N = R_{Base}B_{RW}/M_{FW}$ = 49.5 nXMR / byte
$f_L = 4f_N$ = 198 nXMR / byte
$f_M = 16f_N$ = 792 nXMR / byte
$f_P = 2R_{Base}/M_{NW} = 2f_NM_{FW}/(B_{RW}M_{NW})$ = 200 nXMR / byte
$f_H = 64f_N\max(1, M_{FW}/(32B_{RW}M_{NW}))$ = 3170 nXMR / byte