Scaling Definitions

1) Consensus (Old)

Define

T_R = 3000 bytes ; Reference Transaction weight. Note: T_R must be greater than T_2. T_2 equals the weight in bytes of a 2 input and 2 output transaction.

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 M_L of previous 100001 block (currently = Z_M); Long Term Median

Z_M ; 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 (January 2021)

1) Consensus (New)

Define

T_R = 3000 bytes ; Reference Transaction weight. Note: T_R must be greater than T_2. T_2 equals the weight in bytes of a 2 input and 2 output transaction.

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 M_L of previous 100001 block (currently = Z_M); Long term median

M_L ; Penalty free zone. This is now dynamic.

M_S = the median over the last 100 blocks of max(M_B , M_L) ; 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 $B \leq 0$

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

2a) 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 $F_T = R_{Base}B_T^2$

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

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 the fee required to pay the penalty incurred is the minimum fee.

$B_R = T_R / Z_M$

$F_R = R_{Base}B_R^2$; 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 at $M_F = Z_M$ scaled with M_F

$f_I = 0.2f_R$; Minimum fee per byte

Examples:

T_R = 3000 bytes, R_{Base} = 1.2 XMR, Z_M = 300000 bytes

M_N = 300000 bytes, M_L = 300000 bytes
$f_I = 0.2*R_{base}B_R/M_F$ = 8.00 nXMR / byte ,

M_N = 1425000 bytes, M_L = 1425000 bytes
$f_I = 0.2*R_{base}B_R/M_F$ = 1.68 nXMR / byte

M_N = 1500000 bytes, M_L = 1500000 bytes
$f_I = 0.2*R_{base}B_R/M_F$ = 1.60 nXMR / byte

T_R = 3000 bytes, R_{Base} = 0.6 XMR, Z_M = 300000 bytes

M_N = 300000 bytes, M_L = 300000 bytes
$f_I = 0.2*R_{base}B_R/M_F$ = 4.00 nXMR / byte ,

M_N = 1425000 bytes, M_L = 1425000 bytes
$f_I = 0.2*R_{base}B_R/M_F$ = 0.84 nXMR / byte

M_N = 1500000 bytes, M_L = 1500000 bytes
$f_I = 0.2*R_{base}B_R/M_F$ = 0.80 nXMR / byte

2a) 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 $F_T = R_{Base}B_T^2$

$M_F = \min(M_N , M_L)$ (= M_L if the proposed consensus change is implemented) ; Median for minimum fee calculation

To calculate the minimum fee we consider a transaction of weight T_R at the start of the penalty, B = 0 with $M_N = M_F$ 95% of the fee required to pay the penalty incurred is the minimum fee.

$B_{RL} = T_R / M_F$; Note change M_F instead of Z_M

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

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

$f_I = 0.95f_R$; Minimum fee per byte

Examples:

T_R = 3000 bytes, R_{Base} = 1.2 XMR, Z_M = 300000 bytes

M_N = 300000 bytes, M_L = 300000 bytes
$f_I = 0.95*R_{base}B_{RL}/M_F$ = 38.0 nXMR / byte

M_N = 1425000 bytes, M_L = 1425000 bytes
$f_I = 0.95*R_{base}B_{RL}/M_F$ = 1.68 nXMR / byte

M_N = 1500000 bytes, M_L = 1500000 bytes
$f_I = 0.95*R_{base}B_{RL}/M_F$ = 1.52 nXMR / byte

T_R = 3000 bytes, R_{Base} = 0.6 XMR, Z_M = 300000 bytes

M_N = 300000 bytes, M_L = 300000 bytes
$f_I = 0.95*R_{base}B_{RL}/M_F$ = 19.0 nXMR / byte

M_N = 1425000 bytes, M_L = 1425000 bytes
$f_I = 0.95*R_{base}B_{RL}/M_F$ = 0.84 nXMR / byte

M_N = 1500000 bytes, M_L = 1500000 bytes
$f_I = 0.95*R_{base}B_{RL}/M_F$ = 0.76 nXMR / byte

2b) 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

Z_M ; Penalty free zone for wallet fees

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

$B_R = T_R / Z_M$

$F_N = R_{Base}B_R^2$; 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.

2b) 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_{LW} of previous 100001 block (currently = Z_M); Effective long term median for wallet fees

M_{LW} ; Penalty free zone for wallet fees

M_{SW} = the median over the "last" 100 blocks of max(M_{BW} , M_{LW}); Effective short term median for wallet fees

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

$M_{FW} = \min(M_{NW}, M_{LW})$ (= M_{LW} if the proposed consensus change is implemented) ; Median for wallet fee calculation

$B_{RLW} = T_R / M_{FW}$; Used for the low and normal fees

$B_R = T_R / Z_M$; Used for the medium and high fees

$F_L = R_{Base}B_{RLW}^2$; Low transaction fee for reference transaction

$f_L = R_{Base}B_{RLW}/M_{FW}$; Low transaction fee per byte for a given M_{FW}

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

$f_M = 16\, R_{Base}B_R/M_{FW}$;Medium Transaction fee per byte for a given M_{FW}

$f_P = 2R_{Base}/M_{NW} = f_MM_{FW}/(8B_RM_{NW})$; Maximum Penalty (B =1) Transaction fee per byte for a given M_{NW}

$f_H = 4f_M\max(1, M_{FW}/(32B_RM_{NW}))$; High Transaction fee per byte

Wallet Fee Examples

T_R = 3000 bytes, R_{Base} = 0.6 XMR, Z_M = 300000 bytes

M_{NW} = 300000 bytes, M_L = 300000 bytes
$f_N = R_{Base}B_R/M_{FW}$ = 20.0 nXMR / byte
$f_L = 0.2f_N$ = 4.00 nXMR / byte
$f_M = 5f_N$ = 100 nXMR / byte
$f_P = 2R_{Base}/M_{NW} = 2f_NM_{FW}/(B_RM_{NW})$ = 4000 nXMR / byte
$f_H = f_P$ = 4000 nXMR / byte

M_{NW} = 15000000 bytes, M_L = 300000 bytes
$f_N = R_{Base}B_R/M_{FW}$ = 20.0 nXMR / byte
$f_L = 0.2f_N$ = 4.00 nXMR / byte
$f_M = 5f_N$ = 100 nXMR / byte
$f_P = 2R_{Base}/M_{NW} = 2f_NM_{FW}/(B_RM_{NW})$ = 80 nXMR / byte
$f_H = f_P$ = 80 nXMR / byte

M_{NW} = 1425000 bytes, M_L = 1425000 bytes
$f_N = R_{Base}B_R/M_{FW}$ = 4.21 nXMR / byte
$f_L = 0.2f_N$ = 0.842 nXMR / byte
$f_M = 5f_N$ = 21.1 nXMR / byte
$f_P = 2R_{Base}/M_{NW} = 2f_NM_{FW}/(B_RM_{NW})$ = 842 nXMR / byte
$f_H = f_P$ = 842 nXMR / byte

M_{NW} = 1500000 bytes, M_L = 1500000 bytes
$f_N = R_{Base}B_R/M_{FW}$ = 4.00 nXMR / byte
$f_L = 0.2f_N$ = 0.80 nXMR / byte
$f_M = 5f_N$ = 20 nXMR / byte
$f_P = 2R_{Base}/M_{NW} = 2f_NM_{FW}/(B_RM_{NW})$ = 800 nXMR / byte
$f_H = f_P$ = 800 nXMR / byte

M_{NW} = 75000000 bytes, M_L = 1500000 bytes
$f_N = R_{Base}B_R/M_{FW}$ = 4.00 nXMR / byte
$f_L = 0.2f_N$ = 0.80 nXMR / byte
$f_M = 5f_N$ = 20 nXMR / byte
$f_P = 2R_{Base}/M_{NW} = 2f_NM_{FW}/(B_RM_{NW})$ = 16 nXMR / byte
$f_H = f_P$ = 16 nXMR / byte

Wallet Fee Examples

T_{RW} = 3000 bytes, R_{Base} = 0.6 XMR, Z_M = 300000 bytes

M_{NW} = 300000 bytes, M_{LW} = 300000 bytes
$f_L = R_{Base}B_{RLW}/M_{FW}$ = 20.0 nXMR / byte
$f_N = 4f_L$ = 80 nXMR / byte
$f_M = 16\,R_{Base}B_R/M_{FW}$ = 320 nXMR / byte
$f_P = 2R_{Base}/M_{NW} = f_MM_{FW}/(8B_RM_{NW})$ = 4000 nXMR / byte
$f_H = 4f_M\max(1, M_{FW}/(32B_RM_{NW}))$ = 4000 nXMR / byte

M_{NW} = 15000000 bytes, M_{LW} = 300000 bytes
$f_L = R_{Base}B_{RLW}/M_{FW}$ = 20.0 nXMR / byte
$f_N = 4f_L$ = 80 nXMR / byte
$f_M = 16\,R_{Base}B_R/M_{FW}$ = 320 nXMR / byte
$f_P = 2R_{Base}/M_{NW} = f_MM_{FW}/(8B_RM_{NW})$ = 80 nXMR / byte
$f_H = 4f_M\max(1, M_{FW}/(32B_RM_{NW}))$ = 1280 nXMR / byte

M_{NW} = 1425000 bytes, M_{LW} = 1425000 bytes
$f_L = R_{Base}B_{RLW}/M_{FW}$ = 0.886 nXMR / byte
$f_N = 4f_L$ = 3.55 nXMR / byte
$f_M = 16\,R_{Base}B_R/M_{FW}$ = 67.4 nXMR / byte
$f_P = 2R_{Base}/M_{NW} = f_MM_{FW}/(8B_RM_{NW})$ = 842 nXMR / byte
$f_H = 4f_M\max(1, M_{FW}/(32B_RM_{NW}))$ = 842 nXMR / byte

M_{NW} = 1500000 bytes, M_{LW} = 1500000 bytes
$f_L = R_{Base}B_{RLW}/M_{FW}$ = 0.800 nXMR / byte
$f_N = 4f_L$ = 3.20 nXMR / byte
$f_M = 16\,R_{Base}B_R/M_{FW}$ = 64.0 nXMR / byte
$f_P = 2R_{Base}/M_{NW} = f_MM_{FW}/(8B_RM_{NW})$ = 800 nXMR / byte
$f_H = 4f_M\max(1, M_{FW}/(32B_RM_{NW}))$ = 800 nXMR / byte

M_{NW} = 75000000 bytes, M_{LW} = 1500000 bytes
$f_L = R_{Base}B_{RLW}/M_{FW}$ = 0.800 nXMR / byte
$f_N = 4f_L$ = 3.20 nXMR / byte
$f_M = 16\,R_{Base}B_R/M_{FW}$ = 64.0 nXMR / byte
$f_P = 2R_{Base}/M_{NW} = f_MM_{FW}/(8B_RM_{NW})$ = 16 nXMR / byte
$f_H = 4f_M\max(1, M_{FW}/(32B_RM_{NW}))$ = 256 nXMR / byte

The difference is that the low fee always allows for scaling of the reference transaction, and is the lowest fee that allows for scaling of the reference transaction. Before the lowest fee that allowed for scaling of the reference transaction was the normal fee which was higher than the minimum required for scaling above 300000 bytes.

3) Reference transaction, T_R, Minimum penalty free zone, Z_M, and Median calculation after a fork.

Note: I am not recommending that we reduce T_R from the current 3000 bytes until, at least, we finalize the transaction size for Triptych or another replacement for CLSAG, if Triptych is not used. This is to reduce the impact on fees overall. For example: A reduction in fees followed by an increase in fees or the need to increase Z_M more than would be necessary.

T_{2C} = Weight in bytes of a 2 input and 2 output transaction for the current Monero version rounded up to the nearest 100 bytes. T_{2P} = Weight in bytes of a 2 input and 2 output transaction for the previous Monero version 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 in both versions with the lowest number of inputs and outputs greater than or equal to 2 is used instead.

Calculation of T_R

$T_{RP} = T_R$ of previous Monero version
T_{RC} of current Monero version is set as follows: $T_{RC} \geq T_{2C}$
Recommendation (See: Note) $T_{RC} = \max(T_{2C}, T_{RP})$
$T_R = T_{RC}$ (= 3000 bytes for BP+ fork if ring size < 26)

Calculation of Z_M

$Z_{MP} = Z_M$ of previous Monero version
$Z_{MC} = \max(Z_{MP}, Z_{MP}T_{RC}/T_{RP})$
$Z_M = Z_{MC}$ (= 300000 bytes if T_R = 3000 bytes)

Calculation of M_L, M_S, M_{LW} and M_{SW} where $M_L > Z_M$, or would be under (ii), and where blocks from a previous Monero version are included in a calculation after the fork.

(i) For $T_{2C} \leq T_{2P}$

M_L, M_S, M_{LW} and M_{SW} are calculated in the usual way with no reduction is size applied to the pre fork blocks

(ii) For $T_{2C} > T_{2P}$

When blocks of previous Monero are included in a calculation after the fork version M_B for the pre fork blocks is replaced by M_{BC}, where $M_{BC} = M_B T_{2C}/T_{2P}$. The current value of Z_M is used, and the starting median for M_L and M_{LW} may also need to be adjusted. This needs to be looked at for each specific fork case. There may be a one time increase in M_L, M_S, M_{LW} or M_{SW} at the fork.

4) 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