APPENDIX
PSEUDO-CODE FOR THE SIMULATED ANNEALING ALGORITHM

```
Data: $T_{start}$, $T_{min}$, $maxtimes$, $T_{descent}$, $\lambda$
Result: $X_{result}$
Initialization;
$X_{result}$=[0 0 0 0];
x=[1 0 0 0];
while (($T > T_{min}$)&($times < maxtimes$)) do
    $X_1$=S(j,:);
    if (j==a) then
        x=change(x, $\lambda$);
        adjust(x,s,j);
    end
    $d_f$=F(M,M1,M2,M3,M4,X1)-F(M,M1,M2,M3,M4,X);
    if (if($d_f$ <0)) then
        $X = X_1$;
        if (F(M,M1,M2,M3,M4,$X_1$)-F(M,M1,M2,M3,M4,$X_{result}$))$<$ 0 then
            $X_{result}$=$X_1$;
        end
        $X_1$=S(j+1,:);
        times=0;
    else
        if (rand()$<$ $e^{d_f/t}$ ) then
            $X = X_1$;
            $X_1$=S(j+1,:);
            times=0
        end
    end
    times=times+1;
    $T = T_{descent} * T$
end
```

Algorithm 1: Simulated annealing algorithm