文章编号:1000-6893(2009)01-0099-05

雨流计数的递归算法

蒋东方
(西北工业大学 自动化学院,陕西 西安 710072)

Recursive Algorithm for Rain-flow Counting

Jiang Dongfang
(School of Automation, Northwestern Polytechnical University, Xi'an 710072, China)

摘 要:给出了疲劳分析中雨流计数法的另一种等效解释。在这个解释的基础上,提出了雨流计数的递归算法。该算法将包含多于一个应力峰值点的任意载荷-时间历程以两个谷值点为界分解成 3 个子段,每个子段用递归方法继续分解成 3 个更短的子段,直至每个子段仅包含一个峰值点或不包含峰值点为止。分解得到的仅包含一个峰值点的任意子段对应一个子循环,其应力的最大值和最小值在递归分解过程中可以同时算出。该算法不需像其他雨流计数算法那样在剔除局部子循环的峰、谷点数据后重新整理载荷-时间历程,也能保留各应力子循环出现的先后次序信息,且最先分离出来的可以是变程最大的应力循环。算法的缺点是不适合实时疲劳损伤计算,且算法在实现时需要计算机为其堆栈分配较多的存储器空间。

关键词:疲劳;递归算法;雨流法;载荷历程;循环计数
中图分类号:V215.5^{+}1 **文献标识码**:A

Abstract: In this article, an alternative equivalent explanation for rain-flow counting in fatigue analysis is introduced, based on which a recursive rain-flow counting algorithm is proposed. The algorithm separates any load-time history with more than one peak into three sub-segments by two valleys; and every sub-segment with more than one peak is again separated into three shorter sub-segments recursively, till every sub-segment contains only one single or no peak. A sub-segment that contains a single peak corresponds to a load cycle whose minimum and maximum values are calculated simultaneously by the recursive procedure. The algorithm need not rearrange load-time history after eliminating the peak and valley values of extracted local cycles as is the case with some other rain-flow counting algorithms. It keeps the load cycle sequence information, and the first cycle extracted by the algorithm may be the one with the maximum range. The disadvantages of the algorithm are that it is not suitable for real-time application and that it requires the computer to assign a large amount of memory to stack in its implementation.

Key words: fatigue; recursive algorithms; rain-flow method; load histories; cycle counting

飞机发动机高压压气机盘是发动机的关键部件,其机械疲劳损伤与发动机使用寿命直接相关[1]。统计机械设备经受的应力循环是计算机械设备疲劳损伤的最有效方法。美国材料与试验协会在其标准 ASTM E1049-85[2] 中介绍了 4 大类循环统计算法,其中雨流计数[3]类方法在实际工作中使用最为广泛[4-8]。部分雨流计数算法得到的是半循环和整循环的混合[2-3],另一些雨流计数算法得到的则全部是整循环[9-12]。而将飞机发动机压气机盘的飞行载荷-时间历程分解成应力载荷整循环的叠加更有利于计算压气机盘的疲劳损伤[6]。

现有文献介绍的雨流计数算法,都是从载荷-时间历程中先分离出局部子循环,而主应力循环,也就是载荷变化幅度最大的子循环,直到计数结束前才能得到[9-12]。对于雨流计数在飞机发动机相关研究中的应用,主应力循环参数尤其重要[6]。为此,这里介绍一种对雨流计数法的等价解释,并在此基础上构造一种具有较多良好特性的雨流计数递归算法。

1 雨流计数法的一种等效解释

对于任意一个应力载荷-时间历程,按二阶段雨流计数法,总可以从该载荷-时间历程的最小谷值点处断开,然后整理成起始点和终止点应力值相同、且起止点应力值在载荷-时间历程中最小的一个等价载荷-时间历程。设图 1 就是一个经整理后的载荷-时间历程。图 1 中用字母标识的点

收稿日期:2008-04-21; **修订日期**:2008-06-16
通讯作者:蒋东方 E-mail: jiangdongfang@nwpu.edu.cn

是实际能记录到的应力峰、谷值点，用数字标识的点是为了下文的说明方便而增添的标注点。

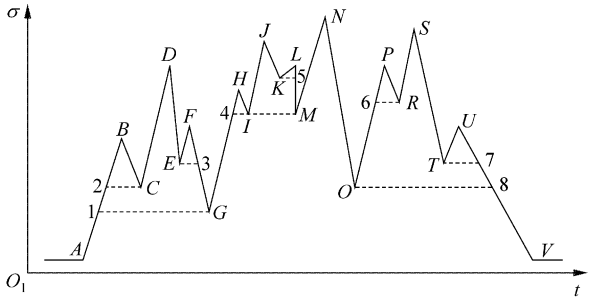

图 1 一个合成的载荷-时间历程

Fig. 1 Synthesized load-time history

对包含 l 个应力循环的载荷-时间历程，雨流计数法的最终处理目标是将该历程分解成一个包含最大交变应力的主循环和 $l-1$ 个叠加在主循环上的次循环。设在图 1 所示历程中，点 N 是该历程的最大峰值应力点，点 A 和点 V 为该历程的起点和终点，并且点 A 和点 V 分别是该历程中最大峰值应力点 N 左、右两边的最小应力谷值点，则该载荷-时间历程的主循环是以 $A1G4IMNO8V$ 组成的应力循环，这个应力循环的应力幅值为点 N 和点 A 的应力差，静态应力为点 N 和点 A 的应力平均值。这个主循环可以等价地看做是雨水从最高应力点 N 向载荷-时间历程左边(起点方向)和右边(终点方向)两个方向流动，遇到下行坡面则沿坡面下行，遇到局部谷点则水平向外平行流动，直到到达应力值最小的起点和终点为止所经过的轨迹构成。这就是雨流计数法的一种等效解释。可以证明，按照这个等价解释对任意载荷-时间历程的应力循环进行计数与四点法对相同载荷-时间历程的循环计数完全相同。根据这个等效解释，在图 1 中，雨水从载荷-时间历程的最大峰值点 N 出发，向左流动时先从点 N 流到点 M，因点 M 不是载荷历程的最低点(点 N 左边最低点应该是点 A)，雨水将平移到点 I，进而平移到点 4，然后继续依次流经点 G、点 1、最后到达点 A 终止。同理，雨水从点 N 向右流动的路线是 $NO8V$。点 N 将以上左、右两条雨流路径连接在一起构成的路线图 $A1G4IMNO8V$，正好是载荷-时间历程所包含的主循环。主循环的最小应力为起点 A 或终点 V 对应的应力，最大应力是峰值点 N 对应的应力。

现将主循环从图 1 中分离出去，留下 $12BCDEF3G$，$4HIJKL5M$，$O6PRSTU78$ 这 3 个子历程部分，同样按上述雨流计数的等效解释，分别对这 3 个子历程进行处理。如对于子历程 $12BCDEF3G$，其左、右最低点分别为点 1 和点 G，其最高峰值点为点 D，从点 D 按等效雨流规律就可以分离出 $12CDE3G$ 这样一个次循环。该次循环的最大应力为点 D 的应力，最小应力为点 G 的应力。切除这个次循环后可以进一步迭代分离出 $2BC$，$EF3$ 等次循环。类似地，可以从子历程 $4HIJKL5M$ 中依次分离出 $4IJK5M$，$4HI$，$KL5$ 等次循环，从子历程 $O6PRSTU78$ 中分离出 $O6RST78$，$6PR$，$TU7$ 等次循环。图 1 中共有 10 个峰值，分解出来的主、次循环数和为 10，与其他任意的整循环雨流计数法[9-12]比较，二者分离出来的子循环的数量相等。

若分别记载荷-时间历程上点 A，N，V 的应力为 σ_A，σ_N，σ_V，载荷-时间历程上其他点的应力类似标记，再记

$$\sigma_{\max}=\sigma_N,\quad \sigma_{\min}=\sigma_A=\sigma_V$$

则图 1 中载荷-时间历程的主循环 $A1G4IMNO8V$ 对应的静态应力 σ_{ave} 和交变应力 σ_{osc} 分别为

$$\left.\begin{aligned}\sigma_{ave}&=(\sigma_{\min}+\sigma_{\max})/2\\ \sigma_{osc}&=(\sigma_{\max}-\sigma_{\min})/2\end{aligned}\right\}\tag{1}$$

从载荷-时间历程中分离出来的其他次循环的静态应力和交变应力的计算类似，如对 $12CDE3G$ 包围的次循环，其静态应力 $\bar{\sigma}_{ave}$ 和交变应力 $\bar{\sigma}_{osc}$ 分别是

$$\left.\begin{aligned}\bar{\sigma}_{ave}&=(\sigma_G+\sigma_D)/2\\ \bar{\sigma}_{osc}&=(\sigma_D-\sigma_G)/2\end{aligned}\right\}\tag{2}$$

以上用雨流法的等效解释对应力载荷-时间历程分解后得到的是完整的子循环，并且能同时得到每个子循环的静态应力 σ_{ave} 和交变应力 σ_{osc}，这些应力参数值与其他能给出应力参数值的雨流计数法[9-12]得出的结果相同。

由于载荷-时间历程的复杂性和随机性，用算法实现以上等效分解方法的最大困难是如何用计算机从任意一个载荷-时间历程中按先主后次的次序正确地分解出各个子循环。

2 递归算法

递归算法的基本思想是将规模较大的问题简化为相同类型的规模较小的问题，逐级迭代直到问题的规模退化到能直接解决为止。

2.1　递归算法举例

阶乘 $f(n)=n!$ 的取值计算是适合用递归算法计算的最典型例子。由于 $f(n)=n\cdot f(n-1)$，阶乘取值的递归计算算法可写为

int Factorial(int n){

(1) 如果($n<0$)则(

① 参数出错;

② return -1;)

(2) 如果($n=0$ 或 $n=1$)则

(return 1;)

(3) (return $n\cdot$ Factorial($n-1$);)

}

以上递归算法只包含 3 条语句，在没有用循环语句的情况下可完成 n 为任意值的阶乘运算，其原因是算法中的语句(3)采用了递归算式。算法中的语句(2)是递归运算的终止条件，即 Factorial(1)=1。

2.2　雨流计数递归算法

雨流法循环计数也可以通过构造适当的递归算法完成。

设全局静态数组 $S[n]$，$N_{start}\leqslant n\leqslant N_{end}$ 是已经经过整理的从起始点 N_{start} 到终止点 N_{end} 的载荷-时间历程记录，且 $S[N_{start}]=S[N_{end}]=S_{min}$。其中 n 包含着应力峰、谷值点在载荷历程中出现的先后次序信息。一种雨流计数的递归算法可以描述为

Void Rain_flow(int N_{start}, int N_{end}, double S_{min}){

(1) 如果($N_{end}-N_{start}<2$)，则(return;)

注解:这种情况说明函数调用时待处理的载荷-时间子历程内无峰值点，故停止对该子历程做进一步的迭代分解，也没有循环计数有关的数据需要输出。

(2) 如果($N_{end}-N_{start}=2$)，则(

① 取 $\sigma_{max}=S[N_{start}+1]$，$\sigma_{min}=S_{min}$;

② 利用式(1)直接计算该单循环的交变应力 σ_{osc} 和静态应力 σ_{ave};

③ 输出 σ_{osc}，σ_{ave} 和($N_{start}+1$);

④ return;)

注解:这种情况说明函数调用时待处理的载荷-时间子历程仅包含一个单峰值，是一个局部应力子循环，因而可以计算并输出该局部子循环的相关数据。其中参数 $N_{start}+1$ 标示了该子循环在载荷-时间历程中出现的先后次序信息。

(3) (

① 在数组 $S[n]$，$N_{start}<n<N_{end}$ 中寻找最大值 S_{max}，记下 N_{peak} 及 $S_{max}=S[N_{peak}]$;

② 若 $N_{start}+1=N_{peak}$，则令 $N_{LV}=N_{start}$，否则，在数组 $S[n]$，$N_{start}<n<N_{peak}$ 中寻找最小值 S_{Lmin}，记下 N_{LV} 及 $S_{Lmin}=S[N_{LV}]$;

③ 若 $N_{peak}+1=N_{end}$，则令 $N_{RV}=N_{end}$，否则，在数组 $S[n]$，$N_{peak}<n<N_{end}$ 中寻找最小值 S_{Rmin}，记下 N_{RV} 及 $S_{Rmin}=S[N_{RV}]$;

④ Rain_flow(N_{LV}, N_{RV}, S_{min});

⑤ Rain_flow(N_{start}, N_{LV}, S_{Lmin});

⑥ Rain_flow(N_{RV}, N_{end}, S_{Rmin});)

}

注解:这一步对应着 $N_{end}-N_{start}\geqslant 3$ 的情况，表明函数调用时待处理的载荷-时间子历程中至少包含两个峰值点，因而需要利用迭代方法继续分解。应注意在语句①处从 $S[n]$，$N_{start}<n<N_{end}$ 中寻找最大值 S_{max} 时，可能出现在 n 的多个取值处，其应力都等于 S_{max}，这时 N_{peak} 取其中任意一个最大峰值应力对应的下标均可。语句②，③处在数组中寻找最小值时遇到类似情况也同样处理。事实上，这一步的语句①是在子历程中寻找最大值峰值点 N_{peak};语句②是在最大峰值点左边寻找与起始点不等同的最小谷值点 N_{LV};语句③是在最大峰值点右边寻找与终止点不等同的最小谷值点 N_{RV}。这 3 条语句找到了将载荷历程分解的两个分界点 N_{LV} 和 N_{RV}，这两个点将从 N_{start} 到 N_{end} 的载荷历程分解成从 N_{LV} 到 N_{RV}、从 N_{start} 到 N_{LV}、从 N_{RV} 到 N_{end} 的 3 个子历程。这一步中语句④～⑥则分别继续对 3 个子历程进行递归计算。调整这 3 条语句的排列次序，可以改变各局部应力循环被分解出来的次序。上文给出的排列次序，称为中序遍历，最先分解出来的是主循环。若将这一步中语句④和语句⑤交换位置，称为左序遍历，则载荷-时间历程中所包含的所有局部应力循环将会按从左到右的次序逐一被分解出来。可见雨流计数的递归算法能保留子循环的次序信息，这个特性也很有用处[13-14]。

3　雨流计数递归算法举例

以图 1 所示载荷历程为例，不妨设 $N_{start}=0$，则 $N_{end}=20$。根据雨流计数递归算法，第 1 次递归运算会直接从算法的第(3)步开始，结果是 N_{peak} 对应图 1 中的点 N，N_{LV} 对应图 1 中的点 G，

N_{RV}对应图 1 中的点 O。从而以点 G 和点 O 为界，将图 1 表示的载荷-时间历程分段为：

Ⓐ 最小应力为 $S_{min}=S[N_{start}]$，包含从点 G 到点 O 之间载荷-时间历程的“序中”子历程；

Ⓑ 最小应力为 $S_{Lmin}=S[N_{LV}]=S[点\ G]$，包含从点 1 到点 G 之间载荷-时间历程的“序左”子历程；

Ⓒ 最小应力为 $S_{Rmin}=S[N_{RV}]=S[点\ O]$，包含从点 O 到点 8 之间载荷-时间历程的“序右”子历程；

算法第(3)步的最后 3 条语句④～⑥又分别对“序中”、“序左”、“序右”子历程进行反复迭代，得到一个如图 2 所示的分解图。

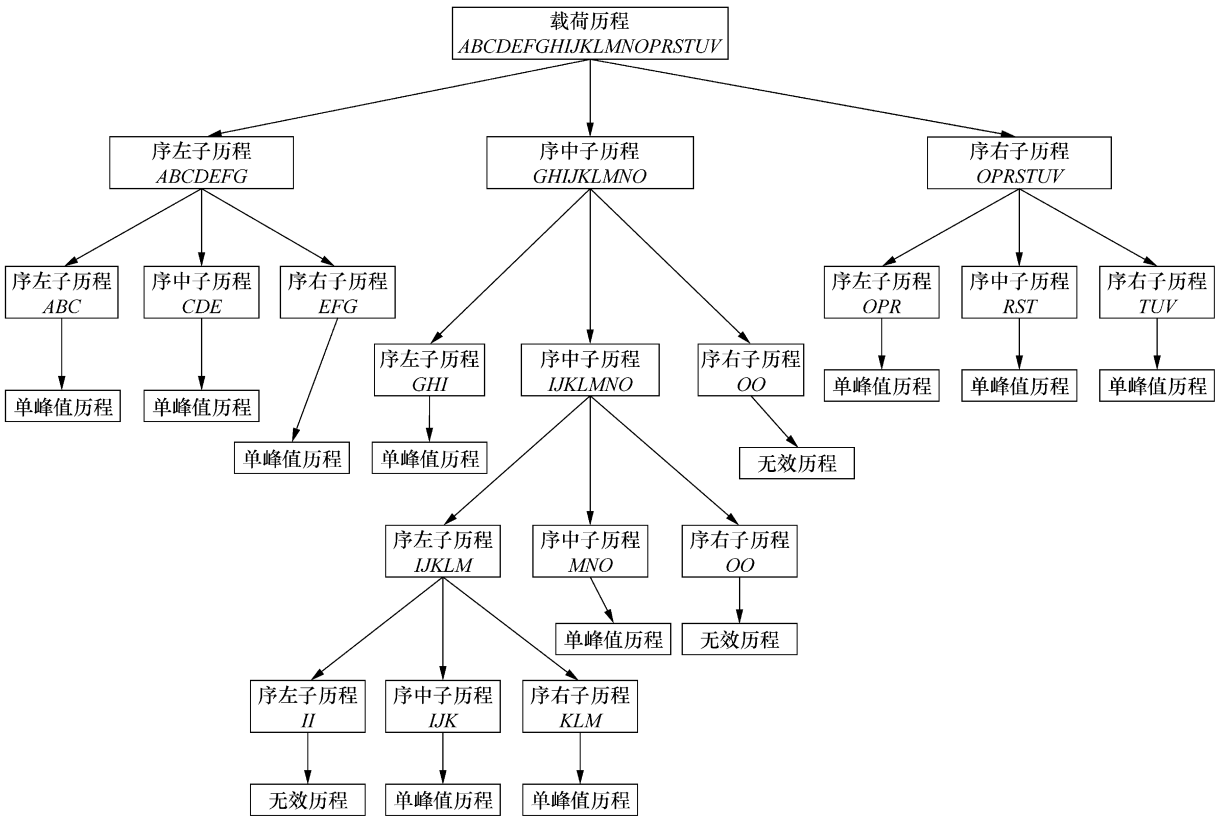

图 2 对一个载荷-时间历程的递归分解示意图

Fig. 2 Recursive disassembled diagram to load-time history

图 2 中将上层父历程分成左、中、右 3 个下层子历程的过程全部由算法的第(3)步迭代产生，图 2 中的“单峰值历程”全部经算法的第(2)步运算终结，图中“无效历程”是递归算法中第(3)-②步中因 $N_{start}+1=N_{peak}$ 或第(3)-③步中因 $N_{peak}+1=N_{end}$ 引入的附加虚历程，递归时被算法的第(1)步清除。从图 2 不难看出，迭代分解算法得到的结果与等效雨流计数法的分析结果一致。

4 算法讨论与结论

所提出的雨流计数递归算法优、缺点并存。优点包括：

(1) 递归算法语法简单，因而易于编程实现。算法中第(2)步输出的($N_{start}+1$)包含着子循环峰值应力在应力数组 $S[n]$中的位置信息，利用该参数可以确定子循环在载荷-时间历程中出现的先后次序；

(2) 递归算法分解出来的子循环全部是完整的局部应力循环；

(3) 采用中序遍历，最大应力循环总是最先被分解出来；

(4) 递归算法在分解出局部子循环后不必对载荷-时间历程数组 $S[n]$进行调整。

雨流计数递归算法的缺点是：

(1) 递归算法在使用前要对载荷-时间历程进行整理(将任意的原始载荷-时间历程从其最小载荷谷点处断开，得到左、右两个子历程片段。如果原载荷循环是闭合的或能通过近似处理整理成闭合的，则可将左段子历程平移到右段子历程后面并与右段相接)，使整理后的载荷-时间历程的

起始点和结束点的载荷相等且载荷值最小，因而不适合实时计算；整理的过程也表明递归算法只适用于处理起止点载荷相等的或能通过近似处理使起止点载荷相等的载荷-时间历程。

（2）递归算法在递归运算过程中需要用堆栈保存递归返回地址及中间计算数据、参数等信息，因而需要计算机有较大的堆栈空间；

（3）递归算法的实现需要用支持递归运算的编程语言完成，如 C，C^{++}，PASCAL 等。

雨流计数的递归算法能准确地完成全循环雨流计数，得到的计数结果与四点雨流计数等方法的结果一致，且能保留子循环在载荷-时间历程中出现的先后次序。在对承受变幅载荷-时间历程的结构部件进行疲劳损伤的非实时计算时，雨流计数递归算法比其他雨流计数方法更具优越性，因而特别适合应用于像飞机发动机转子等具有间歇工作特点的场合。

参 考 文 献

[1] Naeem M, Singh R, Probert D. Implications of engine deterioration for a high-pressure turbine blade's low-cycle fatigue (LCF) life-consumption[J]. Int J Fatigue, 1999, 21(8): 831-847.

[2] ASTM E1049-85 (2005). Standard practices for cycle counting in fatigue analysis[S]. West Conshohocken, Pennsylvania: ASTM International, 2005.

[3] Endo T, Mitsunaga K, Nakagawa H. Fatigue of metals subjected to varying stress—prediction of fatigue lives[C]// The Japan Society of Mechanical Engineers. Preliminary Proceedings of the Chugoku-Shikoku District Meeting. 1967: 41-44.

[4] Sunder R, Seetharam S A, Bhaskaran T A. Cycle counting for fatigue crack growth analysis[J]. Int J Fatigue, 1984, 6(3):147-156.

[5] Glinka G, Kam J C P. Rainflow counting algorithm for very long stress histories[J]. Int J Fatigue, 1987, 9(4): 223-228.

[6] Sunder R. Spectrum load fatigue—underlying mechanisms and their significance in testing and analysis[J]. Int J Fatigue, 2003, 25(9-11): 971-981.

[7] Klemenc J, Fajdiga M. A neural network approach to the simulation of load histories by considering the influence of a sequence of rainflow load cycles[J]. Int J Fatigue, 2002, 24(11): 1109-1125.

[8] Tovo R. Cycle distribution and fatigue damage under broad-band random loading[J]. Int J Fatigue, 2002, 24(11):1137-1147.

[9] Downing S D, Socie D F. Simple rainflow counting algorithms[J]. Int J Fatigue, 1982, 4(1):31-40.

[10] Bannantine J A, Comer J J, Handrock J L. Fundamentals of metal fatigue analysis[M]. Englewood Cliffs, NJ: Prentice-Hall, 1990.

[11] Rychlik I. A new definition of the rainflow cycle counting method[J]. Int J Fatigue, 1987, 9(2):119-21.

[12] Amzallag C, Gerey J P, Robert J L, et al. Standardization of the rainflow counting method for fatigue analysis[J]. Int J Fatigue, 1994, 16(4):287-293.

[13] Anthes R J. Modified rainflow counting keeping the load sequence[J]. Int J Fatigue, 1997, 19(7): 529-535.

[14] Zhu Y G, Zhang S J, Yan M G. A cycle counting method considering load sequence[J]. Int J Fatigue, 1993, 15(5): 407-411.

作者简介：

蒋东方(1968－)　男，博士，副教授。主要研究方向：光电检测技术与仪器，航空电子与自动化装置。

Tel：029-88431357

E-mail：jiangdongfang@nwpu.edu.cn

（责任编辑：徐晓）