CLAIMS

1. A scheduling method for tasks, comprising:

acquiring task information of a current task to be executed, the task information describing the current task to be executed;

determining an actual execution time for the current task to be executed according to the task information; and

comparing the actual execution time with a preset scheduling time corresponding to the current task to be executed, and adjusting an actual scheduling time corresponding to a next task to be executed according to the comparison result so as to determine whether to schedule the next task to be executed,

wherein said adjusting an actual scheduling time corresponding to a next task to be executed according to the comparison result comprises:

determining an excess time when the comparison result is that the actual execution time is greater than the preset scheduling time; and

adjusting the actual scheduling time corresponding to the next task to be executed from the same source according to the excess time to obtain an updated actual scheduling time, and continuing to use it as an actual scheduling time for a subsequent task to be executed from the same source until a subsequent updated actual scheduling time is reduced to 0;

wherein when the current task to be executed is an initial task to be executed from the same source, an actual scheduling time corresponding to the current task to be executed is the same as the actual scheduling time corresponding to the next task to be executed from the same source, and is the preset scheduling time.

2. The method according to claim 1, wherein said acquiring task information of a current task to be executed comprises:

acquiring the task information according to a storage address carried by the current task to be executed.

3. The method according to claim 1, wherein said determining an actual execution time for the current task to be executed according to the task information comprises:

determining an actual execution time for executing the task information as the actual execution time for the current task to be executed.

4. The method according to claim 3, wherein said determining an actual execution time for executing the task information comprises:

inputting the task information into a preset model, and determining an output of the preset model as the actual execution time.

5. The method according to claim 1, wherein said determining an actual execution time for the current task to be executed according to the task information comprises:

matching the task information with multiple pieces of preset task information, and selecting a preset execution time corresponding to matching preset task information according to the matching result as the actual execution time.

6. The method according to claim 1, further comprising:

when the subsequent updated actual scheduling time is 0, determining, upon the arrival of a preset scheduling time corresponding to the subsequent task to be executed, not to schedule the subsequent task to be executed.

7. The method according to claim 1, further comprising:

when the actual scheduling time corresponding to the subsequent task to be executed is not 0 after updated, determining, upon the arrival of a corresponding preset scheduling time, to schedule the subsequent task to be executed according to the actual execution time.

8. The method according to claim 1 or 7, further comprising:

after determining to schedule the next task to be executed, receiving the next task to be executed, and waiting for the arrival of a corresponding preset scheduling time, and then sending the next task to be executed to a corresponding processing device so as to perform task processing according to the corresponding actual execution time.

9. The method according to claim 6, further comprising:

receiving the subsequent task to be executed and, upon the arrival of the corresponding preset scheduling time, sending a task to be executed from other sources to a corresponding processing device so as to perform task processing according to a corresponding actual execution time.

10. The method according to claim 9, further comprising:

waiting for the arrival of a corresponding next preset scheduling time, and then sending the subsequent task to be executed to a corresponding processing device so as to perform task processing according to a corresponding actual execution time,

wherein upon the arrival of the corresponding next preset scheduling time, the actual scheduling time corresponding to the subsequent task to be executed is equal to the preset scheduling time.

11. The method according to claim 1, wherein said adjusting an actual scheduling time corresponding to a next task to be executed according to the comparison result comprises:

determining an idle time when the comparison result is that the actual execution time is less than the preset scheduling time; and

adjusting the actual scheduling time corresponding to the next task to be executed from the same source according to the idle time to obtain an updated actual scheduling time, and continuing to use it as an actual scheduling time for a subsequent task to be executed from the same source.

12. The method according to claim 11, further comprising:

determining to schedule the next task to be executed when the comparison result is that the actual execution time is less than or equal to the preset scheduling time.

13. The method according to claim 11, further comprising:

adding, for the current task to be executed, the idle time to the actual scheduling time corresponding to the current task to be executed so as to update the corresponding actual

scheduling time, and using the updated actual scheduling time as the actual scheduling time for the next task to be executed.

14. The method according to claim 1 or any one of claims 6-11, wherein preset scheduling times corresponding to tasks to be executed from the same source are the same.

15. The method according to claim 1, wherein said acquiring task information of a current task to be executed comprises:

receiving, via a communication interface, the current task to be executed sent by a virtual device.

16. A scheduling method for container tasks, comprising:

receiving a current task to be executed sent by any container, and acquiring task information of the current task to be executed, the task information describing the current task to be executed;

determining an actual execution time for the current task to be executed according to the task information; and

comparing the actual execution time with a preset scheduling time corresponding to the current task to be executed, and adjusting an actual scheduling time corresponding to a next task to be executed from the same container according to the comparison result so as to determine whether to schedule the next task to be executed from the same container,

wherein said adjusting an actual scheduling time corresponding to a next task to be executed according to the comparison result comprises:

determining an excess time when the comparison result is that the actual execution time is greater than the preset scheduling time; and

adjusting the actual scheduling time corresponding to the next task to be executed from the same source according to the excess time to obtain an updated actual scheduling time, and continuing to use it as an actual scheduling time for a subsequent task to be executed from the same source until a subsequent updated actual scheduling time is reduced to 0;

wherein when the current task to be executed is an initial task to be executed from the same source, an actual scheduling time corresponding to the current task to be executed is the same as the actual scheduling time corresponding to the next task to be executed from the same source, and is the preset scheduling time.

17. The method according to claim 16, wherein said receiving a current task to be executed sent by any container comprises:

receiving the current task to be executed according to a communication interface corresponding to the container; and

the method further comprises:

determining a container identifier according to the communication interface so as to determine a task to be executed from the same container according to the container identifier.

18. A scheduling apparatus for tasks, comprising:

an acquisition module for acquiring task information of a current task to be executed, the task information describing the current task to be executed;

a determination module for determining an actual execution time for the current task to be executed according to the task information; and

an adjustment module for comparing the actual execution time with a preset scheduling time corresponding to the current task to be executed, and adjusting an actual scheduling time corresponding to a next task to be executed according to the comparison result so as to determine whether to schedule the next task to be executed,

wherein said adjusting an actual scheduling time corresponding to a next task to be executed according to the comparison result comprises:

determining an excess time when the comparison result is that the actual execution time is greater than the preset scheduling time; and

adjusting the actual scheduling time corresponding to the next task to be executed from the same source according to the excess time to obtain an updated actual scheduling time, and

continuing to use it as an actual scheduling time for a subsequent task to be executed from the same source until a subsequent updated actual scheduling time is reduced to 0;

wherein when the current task to be executed is an initial task to be executed from the same source, an actual scheduling time corresponding to the current task to be executed is the same as the actual scheduling time corresponding to the next task to be executed from the same source, and is the preset scheduling time.

19. A scheduling apparatus for container tasks, comprising:

a receiving module for receiving a current task to be executed sent by any container, and acquiring task information of the current task to be executed, the task information describing the current task to be executed;

a determination module for determining an actual execution time for the current task to be executed according to the task information; and

an adjustment module for comparing the actual execution time with a preset scheduling time corresponding to the current task to be executed, and adjusting an actual scheduling time corresponding to a next task to be executed from the same container according to the comparison result so as to determine whether to schedule the next task to be executed from the same container,

wherein said adjusting an actual scheduling time corresponding to a next task to be executed according to the comparison result comprises:

determining an excess time when the comparison result is that the actual execution time is greater than the preset scheduling time; and

adjusting the actual scheduling time corresponding to the next task to be executed from the same source according to the excess time to obtain an updated actual scheduling time, and continuing to use it as an actual scheduling time for a subsequent task to be executed from the same source until a subsequent updated actual scheduling time is reduced to 0;

wherein when the current task to be executed is an initial task to be executed from the same source, an actual scheduling time corresponding to the current task to be executed is the same as the actual scheduling time corresponding to the next task to be executed from the same source, and is the preset scheduling time.

20. A computing device comprising: a memory and a processor, wherein

the memory is used for storing a computer program; and

the processor is used for executing the computer program for:

acquiring task information of a current task to be executed, the task information describing the current task to be executed;

determining an actual execution time for the current task to be executed according to the task information; and

comparing the actual execution time with a preset scheduling time corresponding to the current task to be executed, and adjusting an actual scheduling time corresponding to a next task to be executed according to the comparison result so as to determine whether to schedule the next task to be executed,

wherein said adjusting an actual scheduling time corresponding to a next task to be executed according to the comparison result comprises:

determining an excess time when the comparison result is that the actual execution time is greater than the preset scheduling time; and

adjusting the actual scheduling time corresponding to the next task to be executed from the same source according to the excess time to obtain an updated actual scheduling time, and continuing to use it as an actual scheduling time for a subsequent task to be executed from the same source until a subsequent updated actual scheduling time is reduced to 0;

wherein when the current task to be executed is an initial task to be executed from the same source, an actual scheduling time corresponding to the current task to be executed is the

same as the actual scheduling time corresponding to the next task to be executed from the same source, and is the preset scheduling time.

21. A computing device comprising: a memory and a processor, wherein

the memory is used for storing a computer program; and

the processor is used for executing the computer program for:

receiving a current task to be executed sent by any container, and acquiring task information of the current task to be executed, the task information describing the current task to be executed;

determining an actual execution time for the current task to be executed according to the task information; and

comparing the actual execution time with a preset scheduling time corresponding to the current task to be executed, and adjusting an actual scheduling time corresponding to a next task to be executed from the same container according to the comparison result so as to determine whether to schedule the next task to be executed from the same container,

wherein said adjusting an actual scheduling time corresponding to a next task to be executed according to the comparison result comprises:

determining an excess time when the comparison result is that the actual execution time is greater than the preset scheduling time; and

adjusting the actual scheduling time corresponding to the next task to be executed from the same source according to the excess time to obtain an updated actual scheduling time, and continuing to use it as an actual scheduling time for a subsequent task to be executed from the same source until a subsequent updated actual scheduling time is reduced to 0;

wherein when the current task to be executed is an initial task to be executed from the same source, an actual scheduling time corresponding to the current task to be executed is the same as the actual scheduling time corresponding to the next task to be executed from the same source, and is the preset scheduling time.

22. A computer-readable storage medium storing a computer program, characterized in that the computer program, when executed by one or more processors, causes the one or more processors to implement steps in the method of any one of claims 1-17.