

Explicit Free Lists: **Freeing a Block**

Insertion policy: Where in the free list do you add a freed block?

LIFO (last-in-first-out) policy

Pro: simple and constant time

Con: studies suggest fragmentation is worse than address ordered

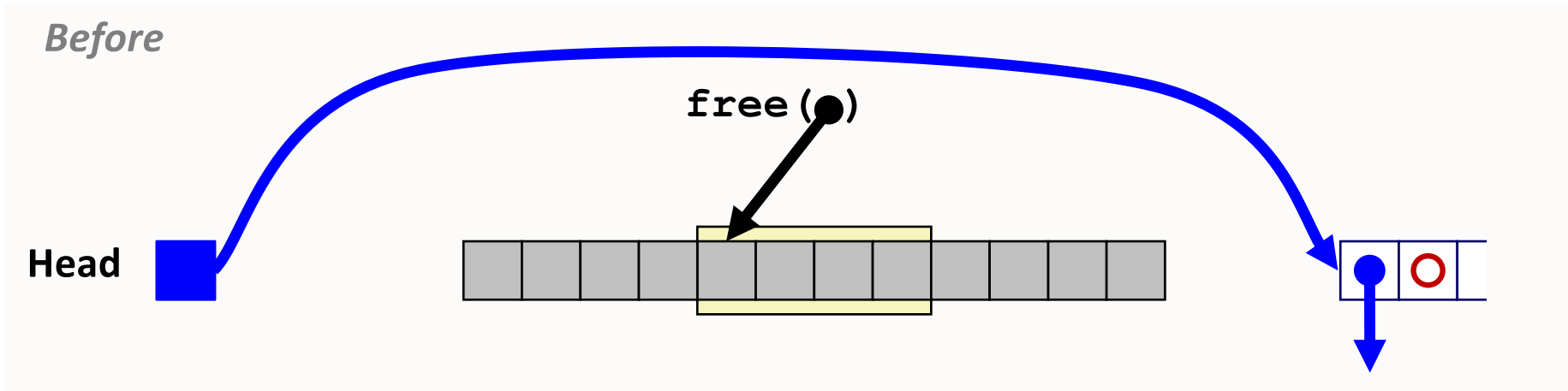
Address-ordered policy

Con: linear-time search to insert freed blocks

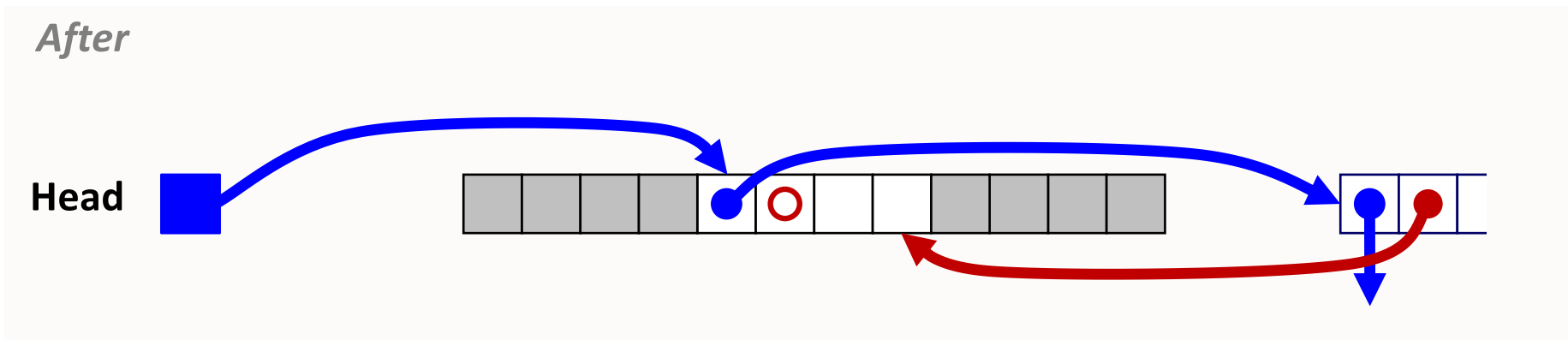
Pro: studies suggest fragmentation is lower than LIFO

LIFO Example: 4 cases of freed block neighbor status.

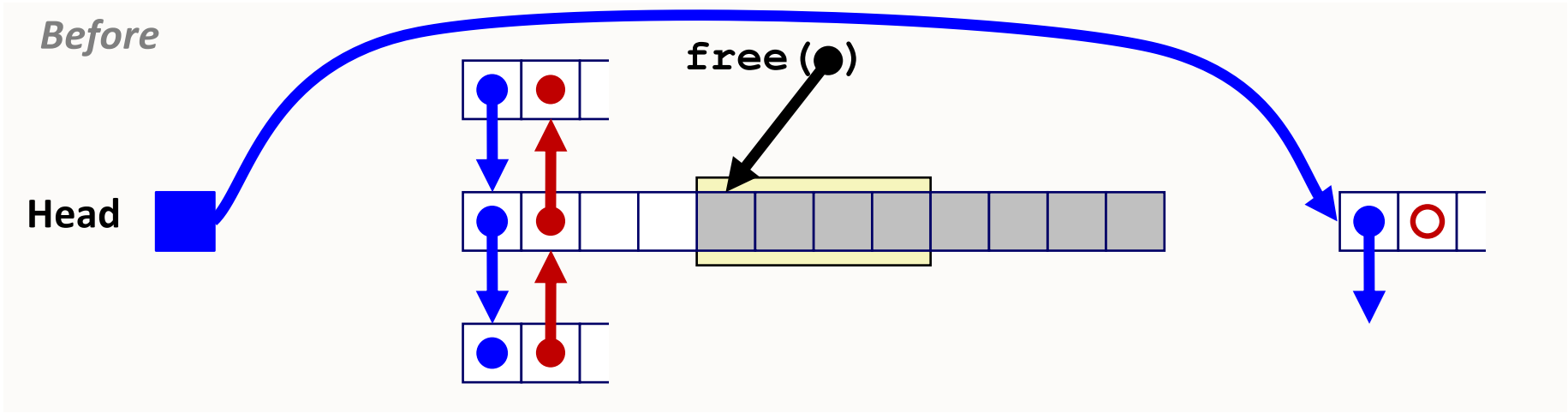
Freeing with LIFO Policy: between allocated blocks



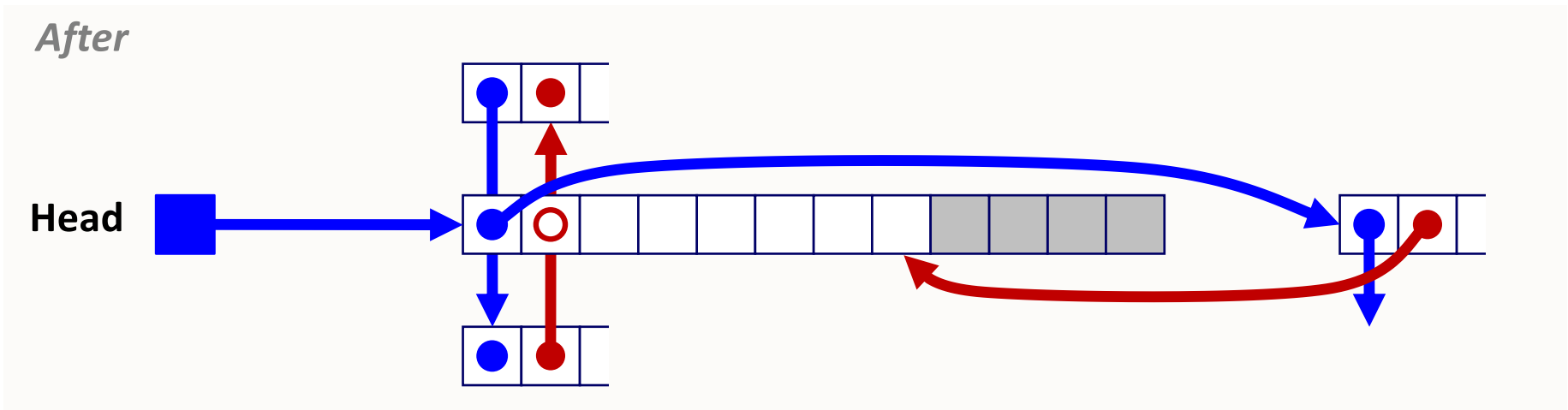
Insert the freed block at head of free list.



Freeing with LIFO Policy: between free and allocated

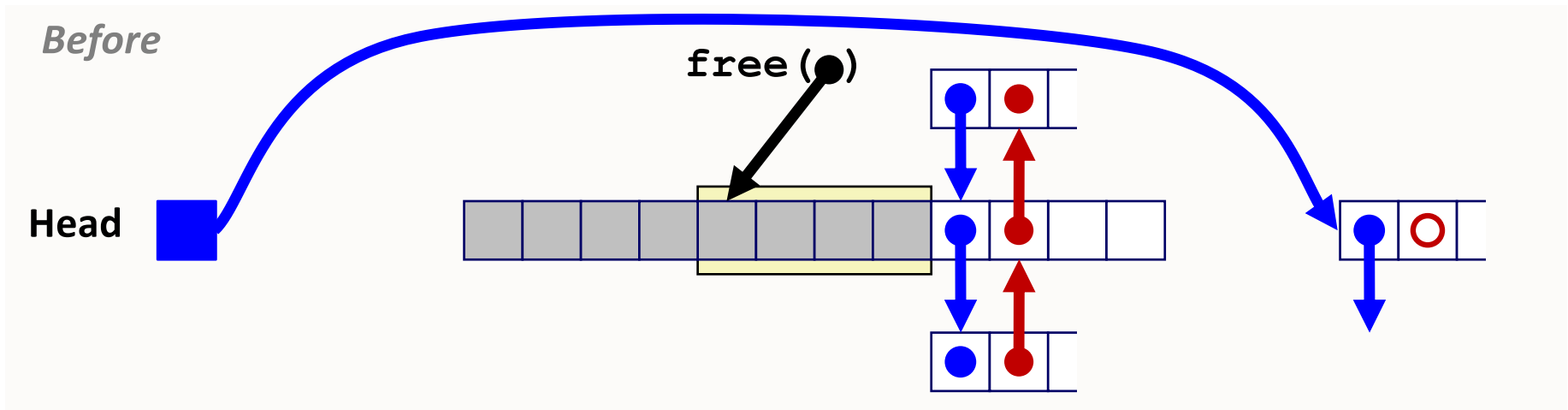


Splice out predecessor block, coalesce both memory blocks, and insert the new block at the head of the free list.

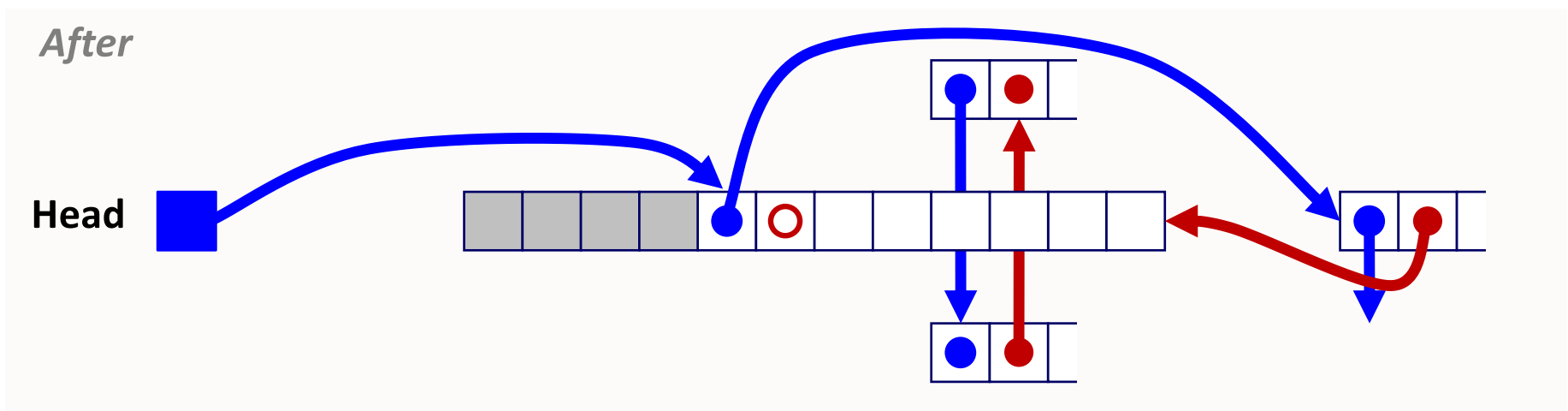


Could be on either or both sides...

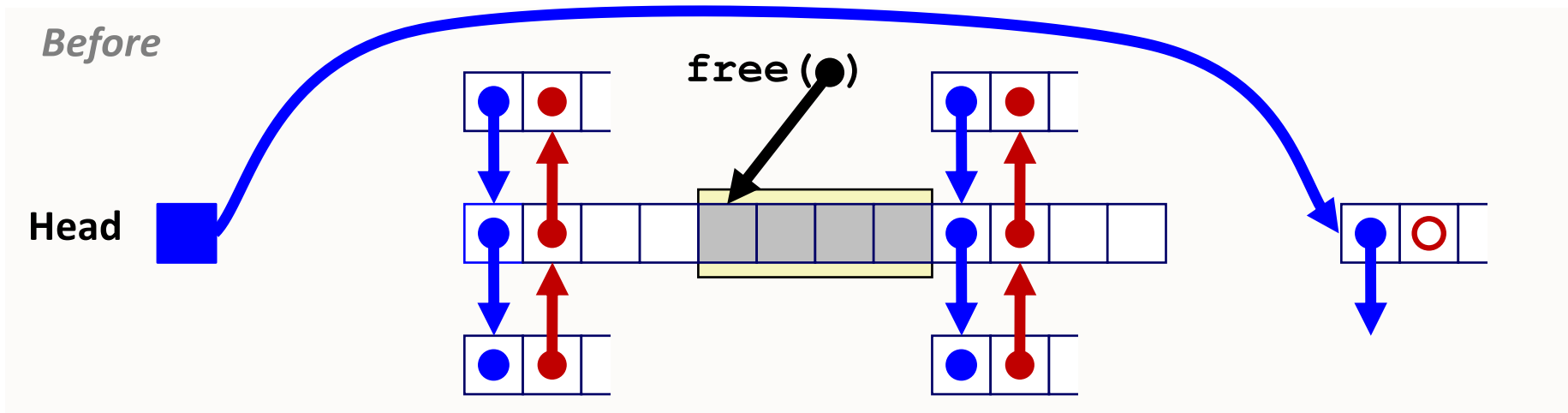
Freeing with LIFO Policy: between allocated and free



Splice out successor block, coalesce both memory blocks and insert the new block at the head of the free list.



Freeing with LIFO Policy: between free blocks



Splice out predecessor and successor blocks, coalesce all 3 memory blocks and insert the new block at the head of the list.

