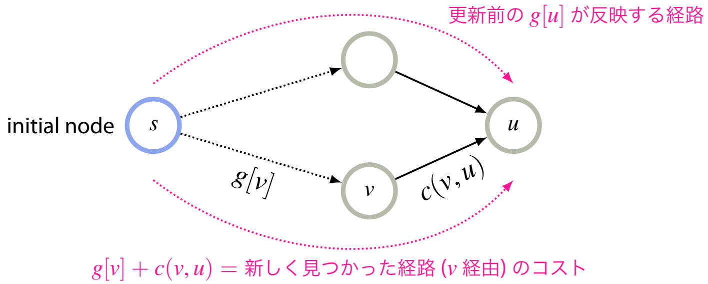
辺 (v, u) に対する “Relaxing” 操作

u における, 出発節点からの最短経路の維持管理



---

if $g[v] + c(v, u) < g[u]$ **then**
  $g[u] \leftarrow g[v] + c(v, u)$ # Update $g[u]$ if a shorter path is found
  $\mathrm{Parent}[u] \leftarrow v$ # Update $\mathrm{Parent}[u]$ to the parent node v

$g[u]$ と “relaxing” 操作の性質

$g^*(u)$ を u への最短 (= 最小コスト) 経路のコストとする

上界性

任意の節点 u について, $g[u] \geq g^*(u)$.
したがって, いったん $g[u] = g^*(u)$ になると, $g[u]$ の値はそれ以上変化しない.

収束性

もし $s \rightsquigarrow v \rightarrow u$ が s から u への最短経路で, (このとき, 必然的に $s \rightsquigarrow v$ もまた s から v への最短経路であることに注意), かつ $g[v] = g^*(v)$, が成り立つなら, (v, u) に “relaxing” 操作を行ったあと, $g[u] = g^*(u)$ が成り立つ.

定理

Dijkstra のアルゴリズムの実行中の任意の時点で, CLOSED 中のすべての節点 v において

$$g[v] = g^*(v)$$

が成り立つ

定理

$g^*(v)$ は s から v への最短経路コストとする.

Dijkstra アルゴリズムを実行中は, 常に以下の関係が成り立っている

CLOSED に含まれるすべての節点 v に関して

$$g[v] = g^*(v)$$

- アルゴリズム開始時, CLOSED は空. なので, 題意はみたされている
- 最初, 出発節点 s が CLOSED に入れられた直後も, 題意はみたされている (なぜなら, s においては $g[s] = g^*(s) = 0$ であるから)

s が CLOSED に入った後に関する証明

背理法を用いる

題意に反し
「ある時点で, $g[u] > g^*(u)$ なる節点 u が CLOSED に入る」

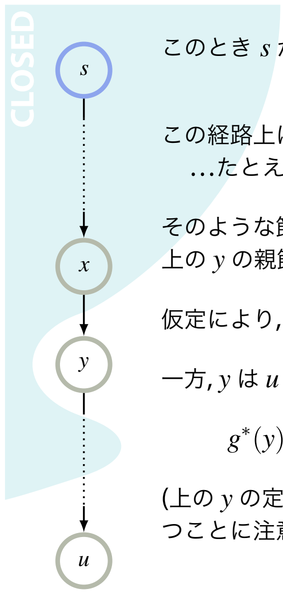
と仮定し, 矛盾を導く

初めてそのような節点 u が CLOSED に入る直前の時点について考える (まだ入っていないことに注意.)

いま, 「u は上の条件をみたす最初の節点である」と仮定したので, この時点 (まだ u が CLOSED に入っていない時点) では, CLOSED 中の任意の節点 v に対して

$$g[v] = g^*(v)$$

が成り立っている.



このとき s から u に至る最短経路に着目

この経路上には CLOSED に入っていない節点が少なくとも一つ存在する
…たとえば u は入っていないので, これは明白

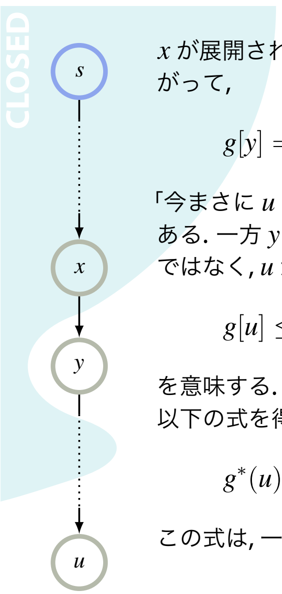
そのような節点のうち, この経路上で s にもっとも近いものを y とし, 同じ経路上の y の親節点を x とする.

仮定により, $x \in$ CLOSED. したがって $g[x] = g^*(x)$

一方, y は u よりもこの最短経路上で前に位置しているので,

$$g^*(y) \leq g^*(u)$$

(上の y の定義だと, $y = u$ の可能性もあるが, その場合にも当然上式は成り立つことに注意)



x が展開された際, 辺 (x, y) に関して “relax” 処理が行われたはずである. したがって,

$$g[y] = g^*(x) + c(x, y) = g^*(y) \qquad (1)$$

「今まさに u が展開されようとしている」という仮定から, u は今 OPEN 上にある. 一方 y の親節点 x が展開されているので y もまた OPEN 上にある. 「y ではなく, u が展開される節点として選ばれた」ということは, すなわち

$$g[u] \leq g[y] \qquad (2)$$

を意味する. 式 (1), (2) と, 背理法の仮定である $g^*(u) < g[u]$ を組み合わせて, 以下の式を得る

$$g^*(u) < g[u] \leq g[y] = g^*(y)$$

この式は, 一つ前のスライドで導いた $g^*(y) \leq g^*(u)$ と矛盾

したがって, 以下の定理が示された

ダイクストラ法を実行している任意の時点で, CLOSED にあるどの節点 v に関しても次式が成り立つ

$$g[v] = g^*(v).$$

(つまり CLOSED 中にあるのは, 最短経路が見つかった節点である)

最短経路がすでに見つかって, そのコストが $g[v]$ に格納されているので, CLOSED にすでに含まれている v の g 値をそれ以上更新する必要はない.

さらに, アルゴリズムは目標節点のどれか $t \in G$ が CLOSED に移されて終了する. t が CLOSED に含まれる, ということは $g[t] = g^*(t)$.