Zadanie 1

Problem ten jest nierozstrzygalny.

Oczwyiście rozważany problem decyzyjny natychmiastowo przeformułowuje się jako: dla danego automatu skończonego $\mathcal{A}$ rozstrzygnąć, czy dla każdych słów $u \in \{a, b\}^*$, $v \in \{c, d\}^*$ istnieje ich taki przeplot, który należy do języka rozpoznawanego przez automat $\mathcal{A}$.

Pokażę, że gdybyśmy umieli rozwiązywać ten problem, to potrafilibyśmy także rozwiązywać problem odpowiedniości Posta. Niech bowiem (α_i, β_i) dla $i = 1, 2, \dots, n$ będzie instancją problemu Posta (tj. pytamy się, czy istnieje skończony ciąg $i_1, \dots, i_k$ taki, że $\alpha_{i_1}\alpha_{i_2}\dots\alpha_{i_k} = \beta_{i_1}\beta_{i_2}\dots\beta_{i_k}$). Bez straty ogólności możemy założyć, że $\alpha_i, \beta_i \in \{c, d\}$, np. kodując wszystkie symbole alfabetu unarnie jako $c^z d$.

Stwórzmy automat $\mathcal{B}$, który będzie akceptował wszystkie słowa postaci: $a^{i_1} b r_1 a^{i_2} b r_2 \dots a^{i_k} b r_k$ takie, że $r_1, \dots, r_k \in \{c, d\}^*$, oraz spełniony jest jeden z warunków:

1. $i_p \notin \{1, 2, \dots, n\}$ dla pewnego p,
2. $|r_p| = |\alpha_{i_p}|$ dla każdego p, lecz dla przynajmniej jednego p ponadto $r_p \neq \alpha_{i_p}$
3. $|r_p| \geqslant |\alpha_{i_p}|$ dla każdego p i dla przynajmniej jednego p ponadto $|r_p| > |\alpha_{i_p}|$
4. $|r_p| \leqslant |\alpha_{i_p}|$ dla każdego p i dla przynajmniej jednego p ponadto $|r_p| < |\alpha_{i_p}|$

a także słowa, w których nie wystąpiło b po ostatnim wystąpieniu a lub nie wystąpiło żadne a.

Ponieważ ciąg (α_i) jest dany, taki automat łatwo zbudować: wybiera on najpierw, który z tych czterech przypadków ma nastąpić, a następnie wiele razy wykonuje operację: wczytuje odpowiednio dużo liter a (nigdy więcej niż n – jeśli wystąpi a^{n+1}, automat od razu akceptuje), wczytuje literę b, po czym czyta następujące słowo r_p, kontrolując jego długość/zawartość ze słowem α_{i_p}. Oczywiście musi on też kontrolować przypadki szczególne, jak np. to, że czy wystąpi jakieś b po ostatnim wystąpieniu a, ale rzecz jasna automaty skończone umieją to robić.

Zauważmy, że pary słów $u \in \{a, b\}^*$, $v \in \{c, d\}^*$, których *żaden* przeplot nie jest akceptowany przez ten automat to dokładnie słowa: $u = a^{i_1} b a^{i_2} b \dots a^{i_k} b$, $v = \alpha_{i_1}\alpha_{i_2}\dots\alpha_{i_k}$. Istotnie, jeśli słowo u nie jest tej postaci (tzn. jest nieprawidłowe syntaktycznie) to np. przeplot uv będzie zaakceptowany. Jeśli więc u jest takiej postaci, to nazwijmy $\hat{v} = \alpha_{i_1}\dots\alpha_{i_k}$. Jeśli $|v| > |\hat{v}|$, to istnieje przeplot, w którym po każdym a^{i_p} dajemy tyle kolejnych liter z v, żeby było ich nie mniej niż $|\alpha_{i_p}|$, oczywiście gdzieś damy ich ściśle więcej. Analogicznie postępujemy jeśli $|v| < |\hat{v}|$. Jeśli $|v| = |\hat{v}|$, to mamy tylko jeden przeplot, który potencjalnie daje akceptację (po każdym $a^{i_p} b$ dać tyle kolejnych liter v, ile wynosi długość α_{i_p}. Akceptacja nastąpi wtedy i tylko wtedy, gdy $v \neq \hat{v}$.

Teraz możemy zbudować automat $\mathcal{C}$ działający tak jak automat $\mathcal{B}$, tylko odnoszący się do słów β_* zamiast α_*. Wtedy automat $\mathcal{A} := \mathcal{B} + \mathcal{C}$ ma następującą własność: pary słów $u \in \{a, b\}^*$, $v \in \{c, d\}^*$, których *żaden* przeplot nie jest akceptowany przez ten automat to dokładnie słowa: $u = a^{i_1} b a^{i_2} b \dots a^{i_k} b$, $v = \alpha_{i_1}\alpha_{i_2}\dots\alpha_{i_k} = \beta_{i_1}\dots\beta_{i_k}$.

Zatem pytanie o to, czy automat $\mathcal{A}$ akceptuje jakiś przeplot dowolnych dwóch słów nad odpowiednimi alfabetami ma taką samą odpowiedź, jak pytanie, czy istnieje rozwiązanie problemu odpowiedniości Posta dla $(\alpha_i, \beta_i)_{i=1}^n$. Zatem gdybyśmy problem z zadania był rozstrzygalny, to problem odpowiedniości Posta także, co nie jest prawdą.

Zadanie 2

Niestety, nie udało mi się rozwiązać tego zadania, więc oczywiście nie liczę na punkty, lecz chciałbym mimo wszystko napisać to, co udało mi się pokazać, a mianowicie wynikanie rozwiązania problemu z zadania z problemu osiągalności dla systemów *Vector Addition System* (VAS), którego rozstrzygalność została udowodniona np. przez Jérôme Leroux (PDF z dowodem znajduje się np. na Pańskiej stronie internetowej, w dziale przedmiotu „Problemy decyzyjne dla systemów nieskończonych").

Definicja VAS, za Leroux: dany jest skończony podzbiór $A \subseteq \mathbb{Z}^d$. Konfiguracją nazwiemy punkt $c \in \mathbb{N}^d$. Między konfiguracjami c i c' można przejść bezpośrednio, jeśli $c' - c \in A$. Problem osiągalności brzmi: dane są dwie konfiguracje c, c', czy można między nimi przejść być może pośrednio, tzn. czy istnieje taki ciąg $c = c_0, c_1, c_2, \dots, c_k = c'$, że $c_i - c_{i-1} \in A$, zaś $c_i \in \mathbb{N}^d$.

Rozpatrzmy bowiem instancję problemu z zadania, tzn. automat skończony z licznikami $\mathcal{A}$, dla którego chcemy roztrzygnąć problem pustości. Oczywiście jeśli zignorujemy litery alfabetu, tzn obłożymy automat homomorfizmem przekształcającym cały alfabet w ε, to uzyskamy automat skończony z licznikami $\mathcal{B}$ z wyłącznie ε-przejściami, dla którego pustość jest równoważna pustości automatu $\mathcal{A}$. Mianowicie: każdy bieg automatu $\mathcal{A}$ przekłada się natychmiast na bieg automatu $\mathcal{B}$ po słowie pustym, i biegi akceptujące w $\mathcal{A}$ to dokładnie biegi akceptujące w $\mathcal{B}$. W drugą stronę, każdy bieg w $\mathcal{B}$ składa się z jakichś krawędzi pochodzących z automatu $\mathcal{A}$. Puszczając automat $\mathcal{A}$ na słowie złożonym z liter na tych krawędziach uzyskujemy słowo, które jest akceptowane przez automat $\mathcal{A}$.

W drugą stronę: dla każdego automatu $\mathcal{B}$ skończonego, z licznikami, z wyłącznie ε-przejściami łatwo budujemy automat $\mathcal{A}$ kładąc na każdym przejściu literę a. Oczywiście istnieje bieg akceptujący $\mathcal{B}$ (siłą rzeczy na słowie pustym) wtedy i tylko wtedy, gdy istnieje jakiś bieg akceptujący automatu $\mathcal{A}$ (na słowie a^n), gdzie n jest długością biegu akceptującego dla $\mathcal{B}$).

Zatem niepustość dla automatów skończonych z licznikami jest równoważna niepustości dla automatów skończonych z licznikami z wyłacznie ε-przejściami. (Liczniki nie odgrywały żadnej roli w tym przejściu, było ono jednie czysto formalną zmianą alfabetu na pusty).

Teraz pokażę, że możemy się pozbyć ograniczenia do jednej operacji $c++$ lub $c--$ na ruch. Rozpatrzmy więc automaty skończone z licznikami i z wyłącznie ε-przejściami lecz mogące w każdm ruchu wykonać więcej niż jedną operację licznikową, tj. relacja przejścia ma postać skończonego podzbioru $Q \times \mathbb{Z}^C \times Q$ – przejście $(q, (r_1, \ldots, r_d), q')$ oznacza, że będąc w stanie q, automat może dodać odpowiednio r_i do każdego licznika c_i (zakładam $C = \{c_1, \ldots, c_d\}$) i przejść do stanu q'. Nazwijmy takie automaty *automatami uogólnionymi.*

Oczywiście model ten jest ogólniejszy od poprzedniego, zatem umiejąc rozstrzygać niepustość dla niego umiemy też rozstrzygać niepustość dla poprzedniego modelu. Trzeba więc pokazać wynikanie w drugą stronę. Załóżmy, że potrafimy rozstrzygać niepustość dla poprzedniego modelu (tj. automatów skończonych z licznikami, z wyłącznie ε-przejściami) i pokażmy, że umiemy rozstrzygać problem niepustości automatów uogólnionych. Rozpatrzmy więc automat uogólniony $\mathcal{C}$. Główna idea polega na zapisaniu przejścia $(q, (r_1, \ldots, r_d), q')$ jako ciągu przejść po kolei inkrementujących lub dekrementujących kolejne liczniki. Można to zrobić, gdyż relacja przejść jest skończona, a zatem możemy zamienić każde takie przejście na $1 + \sum_i |r_i|$ przejść przechodzących po kolei po $\sum_i |r_i|$ nowych stanach. Konstrukcja ta jest dość standardowa, więc nie będę rozpisywał jej formalnie. Oczywiście tak uzyskany automat $\mathcal{B}$ spełnia warunki poprzedniego modelu i istnieje w nim bieg akceptujący po słowie pustym wtedy i tylko wtedy, gdy dla automatu $\mathcal{C}$ istnieje.

Uzyskaliśmy zatem równoważność rozstrzygalności niepustości dla automatów skończonych z licznikiem z rozstrzygalnością niepustości automatów uogólnionych. Teraz uzyskamy równoważność rozstrzygalności niepustości dla automatów uogólnionych z problemem rozstrzygalności niepustości dla automatów uogólnionych bez stanów, które akceptują przez zmianę wartości ustalonego licznika c_f z 0 na 1 (i jest to jedyna operacja jaką wolno im wykonać na liczniku c_0, po jej wykonaniu bieg się kończy).

Oczywiście dla każdego automatu z tego drugiego modelu łatwo wyprodukować automat z pierwszego modelu: automat ma dwa stany: roboczy i akceptację. Przejście, które nie rusza stanu licznika c_f pozostawia automat w stanie początkowym, tj. roboczym, zaś takie, które zmienia stan na 1 przechodzi do stanu akceptującego. Równoważność istnienia biegów akceptujących w obu tych automatach jest oczywista.

W drugą stronę, mając dany automat uogólniony o stanach $q_0, q_1, \ldots, q_n$ zakładamy bez straty ogólności, że jest tylko jeden stan akceptujący q_n – osiągalność jakiegokolwiek stanu akceptującego ze stanu początkowego jest równoważna niepustości. Dorzucamy teraz nowe liczniki $c_{q_0}, c_{q_1}, \ldots, c_{q_n} = c_f$ (oraz pomocnicze $\hat{c}_{q_0}, \ldots, \hat{c}_{q_n}$), z których mówimy, że zawsze jeden ma wartość 1, a reszta ma wartość 0 – ten o wartości 1 odpowiada aktualnemu stanowi automatu. Przejście $(q, (r_1, \ldots, r_d), q')$ zamieniamy na przejście $(r_1, \ldots, r_d), c_q--, \hat{c}_{q'}++$ oraz $(0, 0, \ldots 0), \hat{c}_{q'}--, c_{q'}++$ (z nadużyciem notacji, mieszając oba zapisy zmiany stanów: krotkowy i $++/--$, ale chyba jest to najczytelniejsza postać) – przejście przez $\hat{c}_*$ jest konieczne, aby poprawnie obsługiwać sytuację, gdy $q = q'$. Oczywiście automat ten działa już bez stanów, ale niepustość dla niego jest równoważna niepustości wyjściowego automatu, gdyż biegi akceptujące obu tych automatów są w oczywistej bijekcji.

Do tej pory wszystkie redukcje były równoważnościami, dla następnej (i ostatniej) niestety nie umiem podać odwrotnej. Rozstrzygalność niepustości dla ostatniego modelu wynika z rozstrzygalności osiągalności dla VAS-ów. Rozpatrzmy bowiem dowolny automat $\mathcal{X}$ w ostatnim modelu. Łatwo uzyskujemy automat $\mathcal{Y}$, który zachowuje się tak jak automat $\mathcal{X}$, lecz gdy automat $\mathcal{X}$ wykona podniesienie licznika c_f, automat $\mathcal{Y}$ najpierw zeruje wszystkie pozostałe liczniki – oczywiście nie jest on w stanie sprawdzić, czy stan jest zerowy, lecz może niedeteministycznie wybierać, żeby dekrementować kolejny licznik bądź przejść do kolejnego stanu, w którym będzie dekrementować kolejny licznik etc. Na koniec automat $\mathcal{Y}$ podnosi licznik c_f, tak jak zrobiłby to automat $\mathcal{X}$.

Oczywiście automat $\mathcal{X}$ ma bieg akceptujący wtedy i tylko wtedy, gdy automat $\mathcal{Y}$ ma bieg akceptujący kończący się w stanie, w którym wszystkie liczniki poza q_f są zerami. Jednakże automat $\mathcal{Y}$ jest dobrą instancją VAS – zbiorem wektorów jest relacja przejść, zawierająca obecnie jedynie zmiany dokonywane na poszczególnych współrzędnych. Istnienie biegu akceptującego w $\mathcal{Y}$ jest równoważne osiągalności stanu, w którym wyłącznie $c_f = 1$ ze stanu, w którym wyłącznie $c_{q_0} = 1$, przy wszystkich pozostałych licznikach zerowych.

Zatem z rozstrzygalności problemu osiągalności dla VAS wynika rozstrzygalność problemu z zadania. Łatwo także widać, że jeśli nałożylibyśmy wymaganie, żeby nasze automaty z zadania kończyły w stanie akceptującym, ale z wyzerowanymi licznikami, to problem niepustości dla nich byłby równoważny problemowi osiągalności dla VAS.